# openmediavault Documentation

***Release 8.x.y***

**Volker Theile**

**Jul 12, 2026**

# CONTENTS:

openmediavault is a complete network attached storage (NAS) solution based on Debian Linux.

- It's available for x86-64 and ARM platforms.
- Features a full Web Administration interface.
- Can be controlled via SSH, if enabled.
- Access to file storage is possible with a variety of different protocols such as FTP, SMB/CIFS or NFS and can be controlled with Access Right Management for users and groups.

openmediavault is primarily designed to be used in home environments or small home offices, but is not limited to those scenarios. It is a simple and easy to use out-of-the-box solution that everyone can install and administer without needing expert level knowledge of Networking and Storage Systems.

The system is built on a modular design and can be easily extended with plugins available right after installation of the base system. Additional 3rd-party plugins are available via the OMV-Extras repository.

CHAPTER

ONE

# RELEASES

Table 1: openmediavault historical releases

| Version | Codename | Base Distro | Status | Date Released |
|---|---|---|---|---|
| 0.2 | Ix | Debian 6 | EOL | Oct 2011 |
| 0.3 | Omnious | Debian 6 | EOL | Jul 2012 |
| 0.4 | Fedaykin | Debian 6 | EOL | Sep 2012 |
| 0.5 | Sardoukar | Debian 6 | EOL | Aug 2013 |
| 1.0 | Kralizec | Debian 7 | EOL - Dec 2015 | Sept 2014 |
| 2.0 | Stoneburner | Debian 7 | EOL - Jun 2017 | Jun 2015 |
| 3.0 | Erasmus | Debian 8 | EOL - Jun 2018 | Jun 2016 |
| 4.0 | Arrakis | Debian 9 | EOL - Jun 2020 | Apr 2018 |
| 5.0 | Usul | Debian 10 | EOL - Jun 2022 | Mar 2020 |
| 6.0 | Shaitan | Debian 11 | EOL - Jul 2024 | May 2022 |
| 7.0 | Sandworm | Debian 12 | EOL - Jun 2026 | Mar 2024 |
| 8.0 | Synchrony | Debian 13 | Stable | Dec 2025 |

CHAPTER

TWO

# PREQUISITES

openmediavault is the most flexible Network Attached Storage (NAS) on the market, this is because it relies on running on the operating system Debian GNU/Linux[9]

Any system has software and hardware requirements, also minimal ones and recommended ones.

## 2.1 Hardware requirements

| Item | supported | Minimal | Best | Recommendation |
|---|---|---|---|---|
| DRIV | SSD/HDD/USB… | 1,any | 2,HHD | 2 disks: Seagate Firecuda, WD Black, IronWolf |
| RAM | 1GiB+ any | 1GiB | 4Gig | 8GiB+ dual channel DDR4/DDR3 |
| NIC | WiFi/Ether/USB | any | 10Mb NIC | 1GiB NIC or 10Gb NICs: SFP fiber |
| CPU | arm,x86,x64 | 32bit | 64bit | Intel Dual Core, AMD Ryzen |

### 2.1.1 Storage drives (DRIV)

This is the key of the system. openmediavault supports any drive hardware and any drive interface (SATA, ISE, SCSI, USB, SERIAL) but information and management will depend on the disk drives supported by Debian system. **Supported Debian OS versions can be checked in the** *releases section*.

Modern storage drives have firmware inside that reports several attributes. Any storage drives is supported but openmediavault only supports SMART for those disks that are connected to an HBA in pass-through mode.

The system manages two types of storage unit classification:

**System drive storage (SDS)**
: The Storage Disk Drive(s) used to put the system program files. **This system drive (which can consist of several partitions) cannot officially be used for shared resources or as user data drive**. The recommendation for the system drive(s) hardware could be older spinning Hard Disk Drives (HDD), Disk-on-Module[3], CompactFlash[4] or thumb drives (USB), the Solid State Disks (SDD[2]) must be managed. Best option are the Helium Hard Drives (HHD).

**Data drive storage (DDS)**
: The storage disk drive(s) where the user data will be stored. **Cannot be the same of the system drive(s) and sizes will depend of the usage of the stored user data**. Shared resources are managed over partitions, which is discussed in the software requirements topic below. The recommendation for the Data drives must be Solid State Disks (SSD[2] disks) for best performance or spinning Hard Disk Drives (HDD) for better device durability over time.

[9] https://www.debian.org/intro/about.en.html#what
[3] https://en.wikipedia.org/wiki/Solid-state_drive#DOM
[4] https://en.wikipedia.org/wiki/CompactFlash
[2] https://en.wikipedia.org/wiki/Solid-state_drive

For RAID, SnapRAID, or mergerfs setups, prefer CMR drives over SMR. SMR drives can stall during rebuilds and parity sync, and the recording technology usually isn't obvious from the model number. A reference like the NAS CMR/SMR drive list helps you check before buying.

### 2.1.2 Memory (RAM)

Sufficient RAM is vital to maintaining peak performance. There are several combinations of installation sizes but the **recommendation is at least 4GiB for novice administrators**.

For best practice you should have 8GiB of RAM for basic operations in default installation, and no matter the size, RAM must be configured in Dual Channel mode[8] which will improve performance in low memory installations.

Unlike other NAS systems openmediavault can run in with as low as 1GiB of RAM, but of course will need expertise.

### 2.1.3 Network interface card (NIC)

A NAS system as means is a network oriented system, of course openmediavault is the most flexible NAS system in the world and allows multiple ways of communication, this is because openmediavault offers several modes of shared resources like RSYNC that do not depends on networking. But there are few key points:

- NAS write speed to storage is limited to the speed of your NAS computer network interface.
- NAS write speed to storage is limited to the speed of your network.
- Write speed to storage depends on ratio of the number of users accessing it.

Full-duplex NICs are recommended, half-duplex NICs typically offer 10/100 Mbit/s and storage drives far exceed those speeds but are of course supported by openmediavault.

Any supported NIC vendor is determined by the support of the operating system where openmediavault system resides and runs, which is Debian GNU/Linux.

### 2.1.4 Architecture (CPU)

We have already denoted in the requirements table the supported architectures or processors (which reduces and determines which computers can be used) but this is really a reference since it is determined by the support of the operating system where openmediavault system resides and runs, which is Debian GNU/Linux.

Any computer system supported by the common requirements of Debian operating system could be an openmediavault system installation target. **Currently supported are AMD64 (x86/x86-64 64bits), i386 (x86 32bits), ARMv7+ARMv8 (arm64 64bits armel/armhf 32bits)**. The i386 version is no longer officially supported since version 6, although it is still published by the project due to the nature of openmediavault technology.

## 2.2 Software requirements

| Item | Software | Minimal | Best | Recommendation |
|---|---|---|---|---|
| OS | Debian Linux | 13 | 13 | |
| BOOT | BIOS,UBOOT,UEFI | BIOS,mbr | BIOS, GPT | Disable Secure boot, GPT table |
| SDS | HDD,SSD,USB… | 1, 4GiB | 2, 120+500GiB | Disk drive with 120G root size, 8G swap size |
| DDS | HDD,SSD,USB… | 0 or any | HHD,1 per share | One disk or part per shared resource |
| NET | LAN,WAN,SAN,VPN | LAN | SAN,PAN,LAN | Fiber IPv4, or at least cable LAN |

[8] https://en.wikipedia.org/wiki/Multi-channel_memory_architecture

### 2.2.1 Operating system (OS)

openmediavault is a piece of software, it resides and runs under a key software OS named Debian GNU/Linux as "the universal operating system"[Page 5, 9]

**Supported Debian OS versions can be checked in the** *releases section*.

While possible to deploy in a virtual environment, depending of the nature of virtualization it will degrade the performance access in various ways. Installation in LXC or any other container based solution is not supported.

Whatever the situation, openmediavault assumes that target operating system does not have any previous installation of any of the programs used by openmediavault dependencies:

| software | package | related to | Observations |
|---|---|---|---|
| http server | nginx | web interface | port 80 must be free at installation |
| http system | php | web interface | only OS php packages wil be used! |
| display GUI | lighdm,xdm.. | Desktop install | **Conflicts, any will be deinstalled** |
| network man | netplan.io | Networking | network is managed by openmediavault with netplan |
| ssh server | ssh | Remote access | any configuration will be managed |
| smb server | samba | Shared drives | any configuration will be managed |
| quota man | quota | Quota manage | any configuration will be managed |

### 2.2.2 Device boot (BOOT)

The boot type and its support is determined by the Debian operating system which supports BIOS, UEFI, PXE, UBOOT and many others, the configurations of these depend on the installation of the operating system and once done the openmediavault can be installed without problems.

However the downloadable and ready-to-use ISO image from the official openmediavault website only supports AMD64 with BIOS mode boot at the moment. If you want to have openmediavault on other computers with another boot type/mode, then you should install Debian first and then *manually* openmediavault.

**Supported Debian OS versions can be checked in the** *releases section*.

### 2.2.3 System drive storage (SDS)

Any brand of storage disk will be supported as long as it is supported by Debian since it is where openmediavault system resides and runs, check it at the *releases section*.

The system storage disk will be fully managed and taken over by openmediavault after installed. In case of the official ISO installation, will automatically partition the system storage disk drive into 3 partitions. See table below for details.

In case of a manual installation on a previous Debian operating system, this **drive should have at least two partitions**. See table below for details.

| Partition | Mininmal | Best size | Mandatory |
|---|---|---|---|
| `/boot` | 256Mib | 500Mib | Optional, partition used to boot |
| `/` | 4Gi | 120GiB | Yes, the partition were system install |
| `swap` | 100Mib | 16GiB | Optional, the partition for virtual ram |

### 2.2.4 Data drive storage (DDS)

The data storage disk will be partially managed by the openmediavault after installation, and only in customized installation this can be a partition of the same system drive storage.

Each data storage drive will be managed, but regardless of the partitioning scheme, all data drive partitions will be mounted at the `/srv/` path. The **data storage drive must be a different disk drive than the system disk drive** where the operating system is installed. **One partition of each data drive can handle one or many shared resources.**

| Partition | Min size | Best size | Mandatory |
|---|---|---|---|
| `/srv/<*>` | 100Mib | bigger | Yes, size as need! E.g. 128GiB,500GiB,4TiB |

## 2.3 Technical notes

Take into **consideration that the more customized the installation is, the less supported it could be**.

openmediavault allows for deployments on systems with low hardware/software specification, at the hands of knowledgeable operator while other NAS systems allow ease of use at the cost of high requirements. openmediavault allows both as well in a well balanced use case![7]

### 2.3.1 About minimal or custom setups

Please note that openmediavault systems can be installed on just 4GiB of system partition, with no more than 1GiB of RAM and using a WiFi or USB connection to access it remotely, on any small device such as Raspberry Pi or Banana Pi boars which are ARM, or old i386s machines no matter if those are 32 or 64 bits, but of course as more unique the configuration the more knowledge will be need as per issue comment #131.

Technically OMV can be installed on a single storage disk, this is possible if it is done on a previously configured Debian system, with a free partition apart from the 3 necessary system partitions, previously formatted and configured. Of course this is only possible for skilled Linux users.

### 2.3.2 Drive Storage technical details

System Drives are not managed as same of Data Drives. System drives are not so intensively used, but Data Drives will need tricks to extend the useful life. Drivers support is by the project https://www.smartmontools.org/wiki/TocSupport

If you use a Flash Drive, select one with static wear leveling 6, without this the drive will have a very short lifetime. It is also recommended to install and activate the Flash Memory plugin.

In the same Solid State Disk, or rather Drive (SSD) for x86 based architectures, is usually only recognized properly by the BIOS or UEFI, when in the BIOS/UEFI the feature AHCI has been activated for SATA (instead of IDE). Modern computers have that by default. But on old machines the default might/could be IDE. About ARM based computers this is not a problem, when used SATA interfaces, but on eMMCs will need some tuning because they are treated like Flash drives.

On SSDs, the cleaning action TRIM is recommended for sustained long-term performance. Otherwise it might become slow with time. Very old SSD's from before 2010 usually don't support TRIM.

Take note that eMMCs, SSDs, Flash drives will have a lifetime degradation, excessive writes wear out those drives faster, specially in they are very cheap.

On SSDs Hibernation (suspend-to-disk) causes a huge amount of write actions, openmediavault is a server system so it is expected to stay always on or off.

### 2.3.3 About compression or encryption

Enabling encryption on SSDs also means more writes which wear out SSDs, eMMCs or Flash drives faster.

As well as enabling compression on filesystems like Btrfs or ZFS; although Ext4 has better commit timing; a parameter with commit=600 to 800 is best for this particular one.

[7] https://forum.openmediavault.org/index.php?board/29-guides/

### 2.3.4 Partition table technical details

Of course, the supported partition tables will depend on the installation mode, and openmediavault can handle any type of partition table supported by the Debian. This is because to manage shares on new or uninitialized storage drives it will need to create partitions or at least read structure of them.

CHAPTER

THREE

# INSTALLATION

**Warning**

openmediavault does not support installation in LXC or any other container based solution.

**Before you begin:**

- Check if your hardware is supported on the system *requirements page*.

**Installation variants:**
Choose your installation variant and follow the instructions.

**Note**

Choose the *Dedicated drive* variant if you like to install openmediavault from scratch on x86/AMD64 hardware. For SBC systems based on the ARM32/64 architecture use the *Debian Operating System* variant.

- *Dedicated drive* - Recommended method via ISO image. This runs openmediavault from its own drive.
- *USB flash drive* - This runs openmediavault from a USB flash drive.
- *Debian Operating System* - Use an existing Debian OS installation for openmediavault.
- Debian Operating System via debootstrap. Use this as a last resort in case the installer does not recognize a specific essential hardware component like hard disk (NVME) or a network card that needs a higher kernel (backport).
- *SD card* - This runs openmediavault from a SD card.

**First time use:**
If you have a screen attached, KVM or IMPI console the login screen will display the current IP address assigned for the web interface. Open your browser and type that IP address. The default web interface login credential are `admin:openmediavault`, the `root` password is the one you setup during installation.

For ARM images the root password is the same as admin password.

**Note**

openmediavault will enable SSH access for the user `root` by default to be able to access a headless system in case of a broken installation or other maintenance situations. You should disable this behaviour in the `Services | SSH` page for security reasons after installation.

To still get `root` access you need to create a non-privileged user and add them to the `_ssh` and `sudo` groups. After that you can SSH into the system with this non-privileged user and run `sudo su`.

## 3.1 Installation via ISO image

**Important**

Disconnect all disk devices except the one that will be used for the operating system before starting the installation.

**Warning**

The operating system/root filesystem will consume the whole space of the picked disk device. All existing data on that disk device will get lost.

**Important**

Ensure that you have a working Internet connection during installation. An offline installation will lead to problems later on.

### 3.1.1 Download

You can down load the ISO images here.

### 3.1.2 Burn the installer

For x86 architecture you can burn the ISO directly into a USB drive using etcher or dd linux utility:

```
$ sudo dd if=xxx.iso of=/dev/sdX bs=4096
```

If you have a CD-DVD burner, you can burn the ISO into an optical media then boot from CD or DVD.

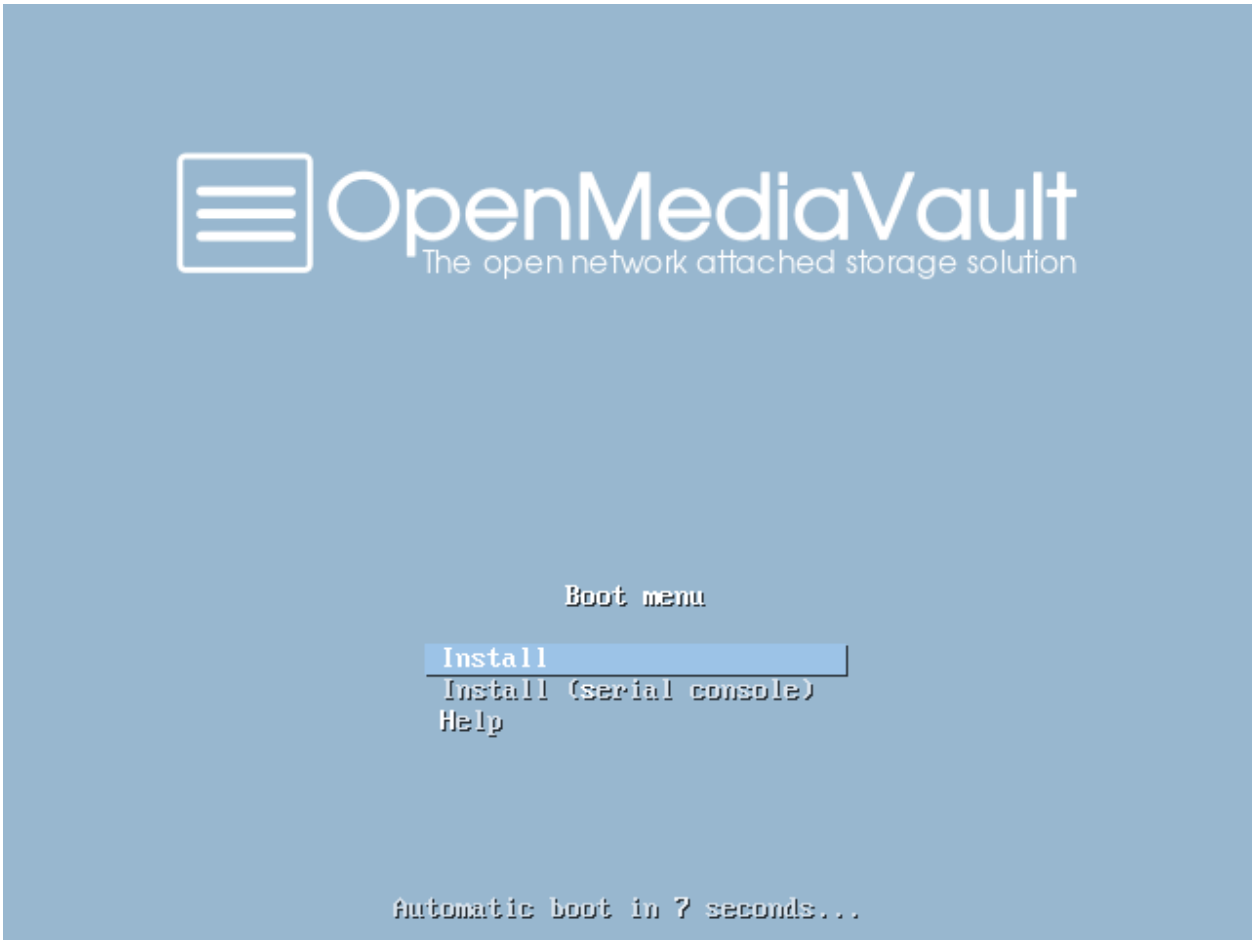
### 3.1.3 Boot the installer

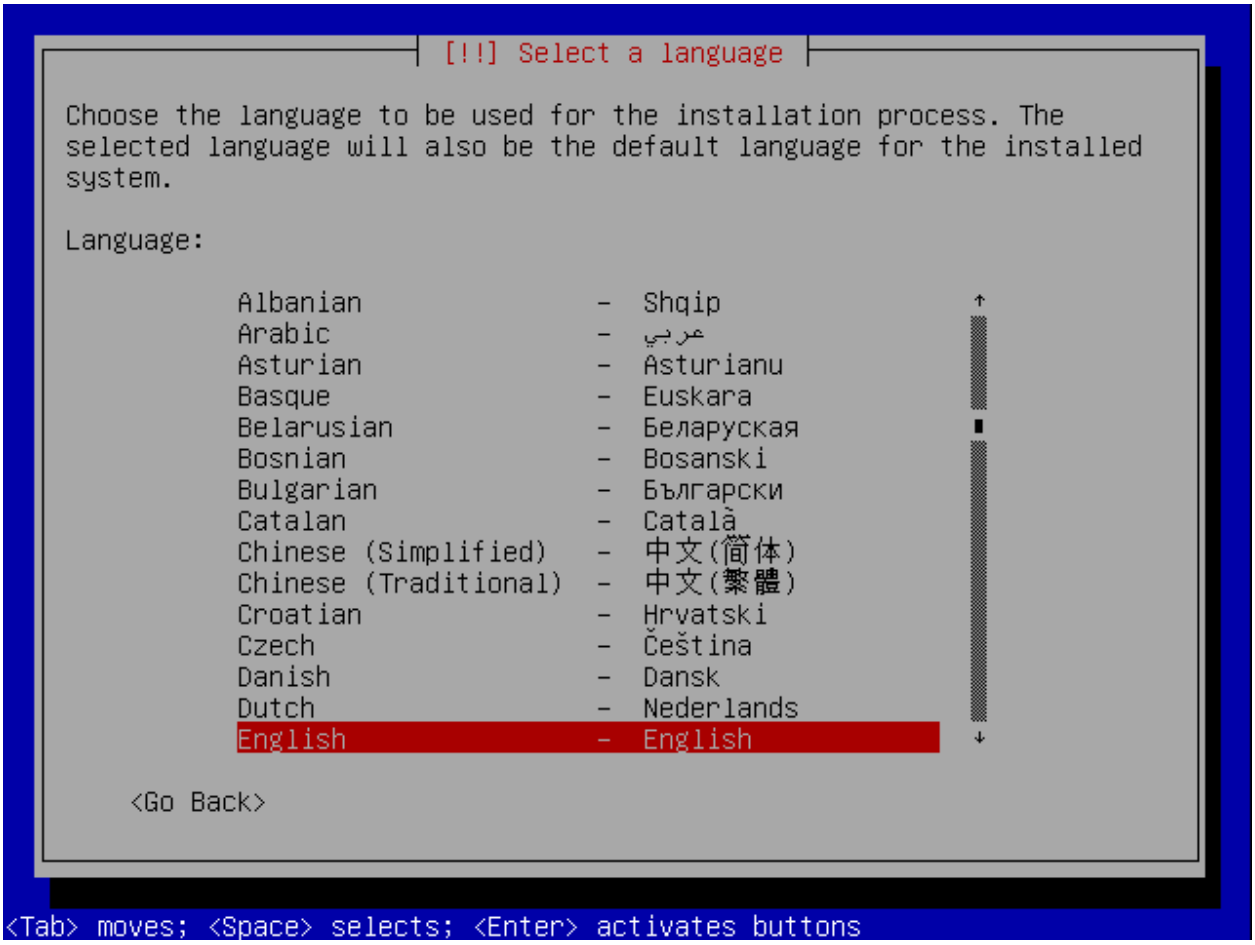
For x86 architecture, enter BIOS configuration, select to boot either from USB or CD and reboot.

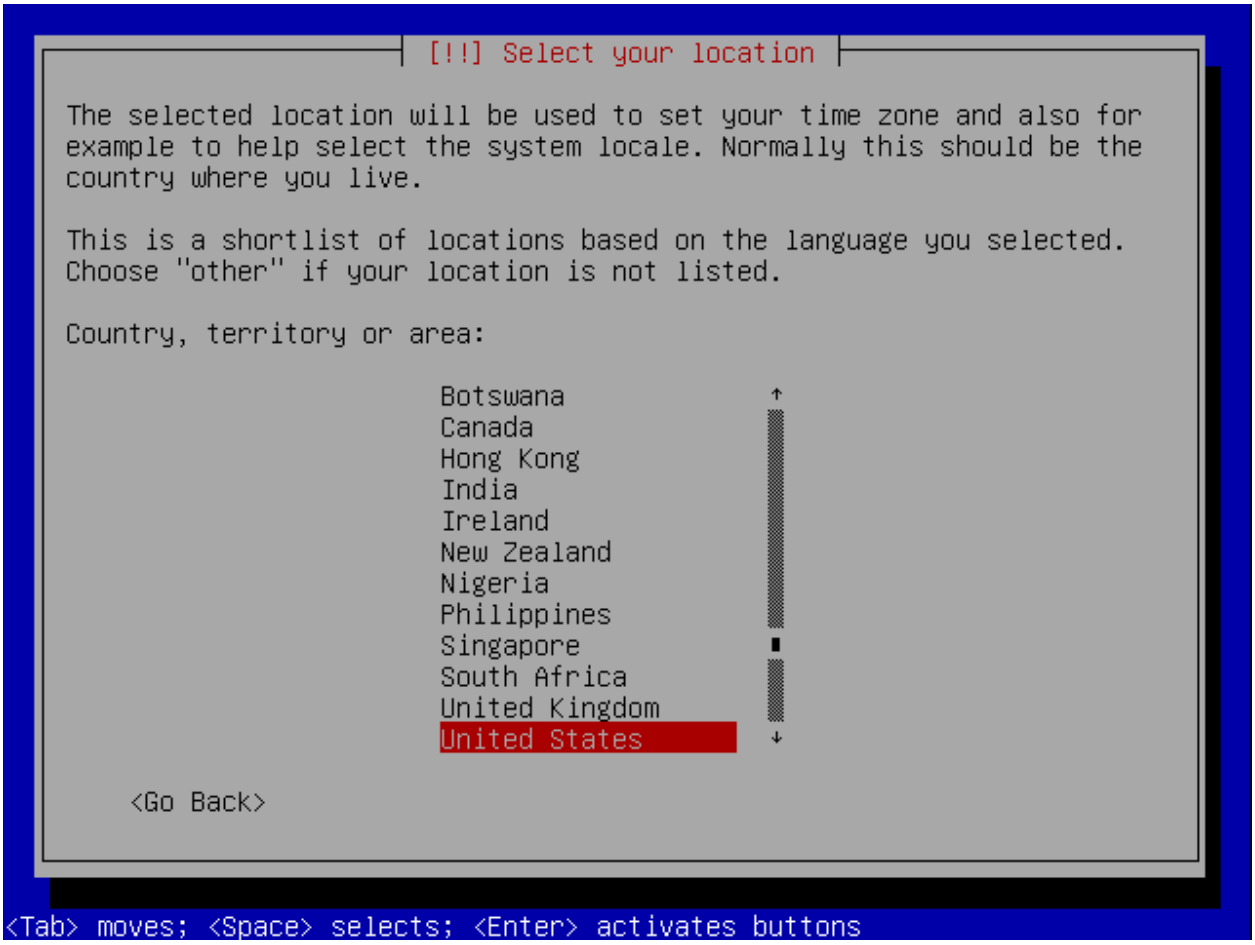
### 3.1.4 Installer

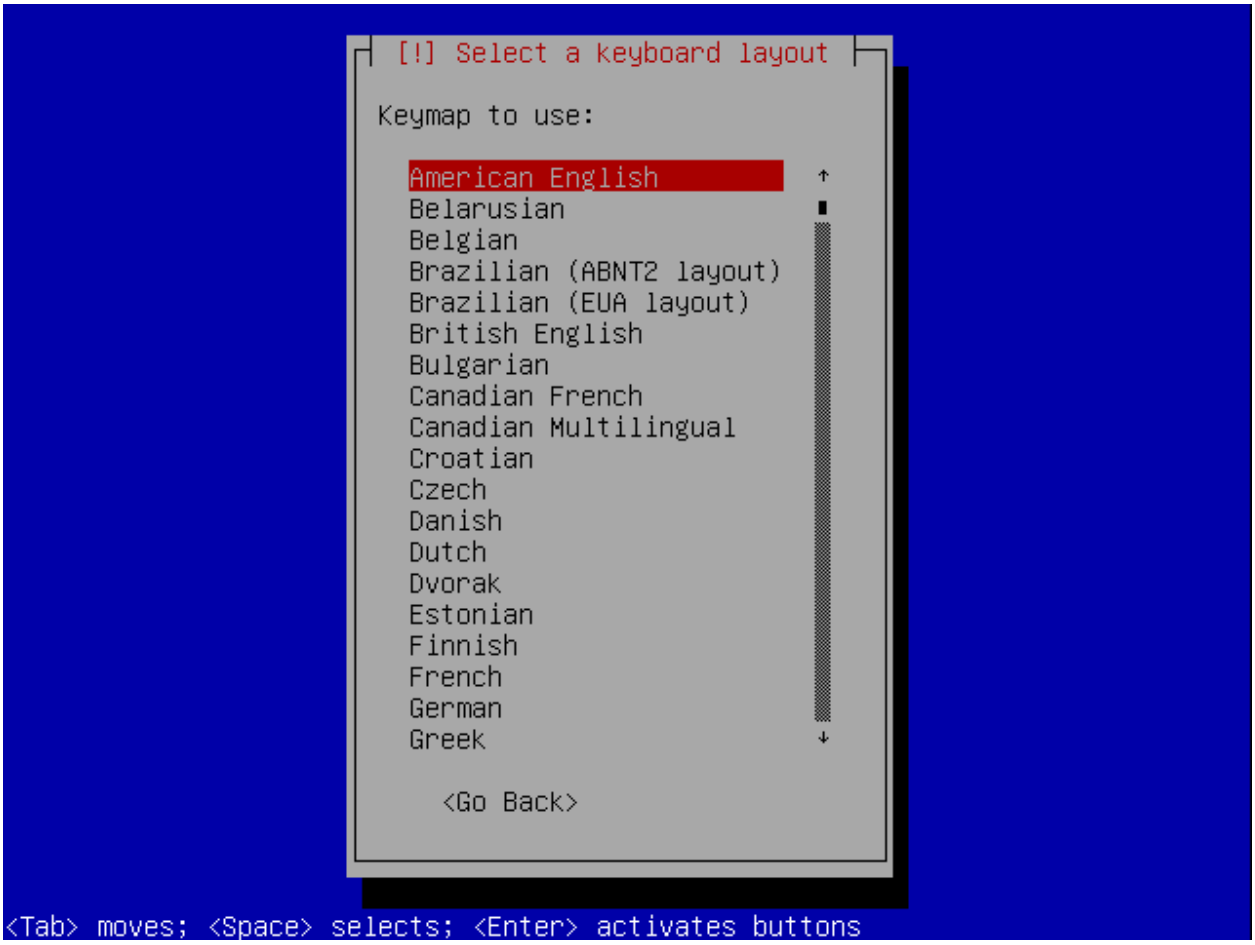
The current ISO installer is reduced to have minimal interaction. You will prompted to select location, language and root password. The installer will pick the first available disk to deploy the the OS. You will create a root password, but you will be able to create other (administrator) accounts for SSH and web interface use after first reboot.

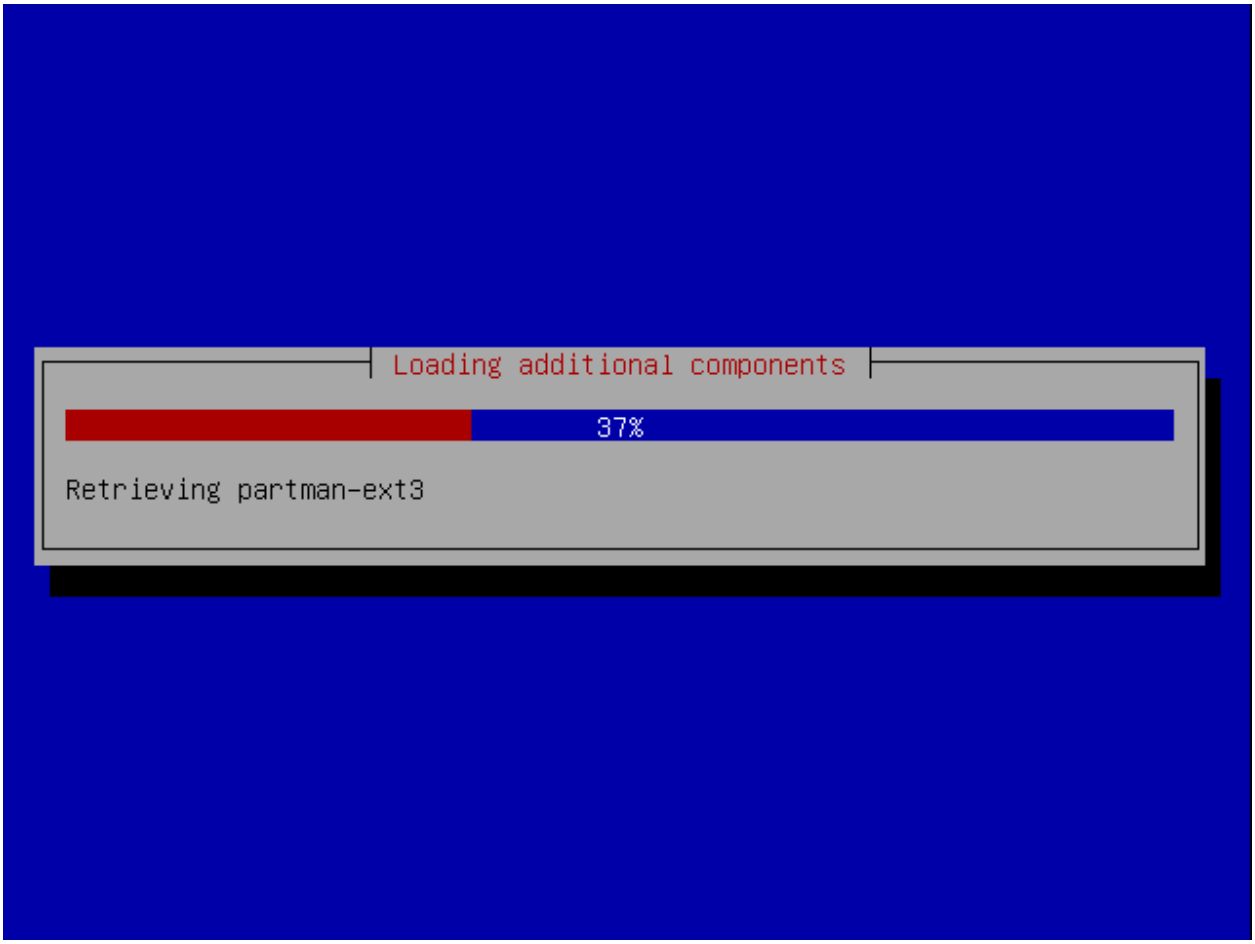
Once the installer finished the system will reboot, make sure you remove the installer and select BIOS to boot from the disk where openmediavault was installed. You can also start connecting any data drives you previously disconnected before install or reinstall.

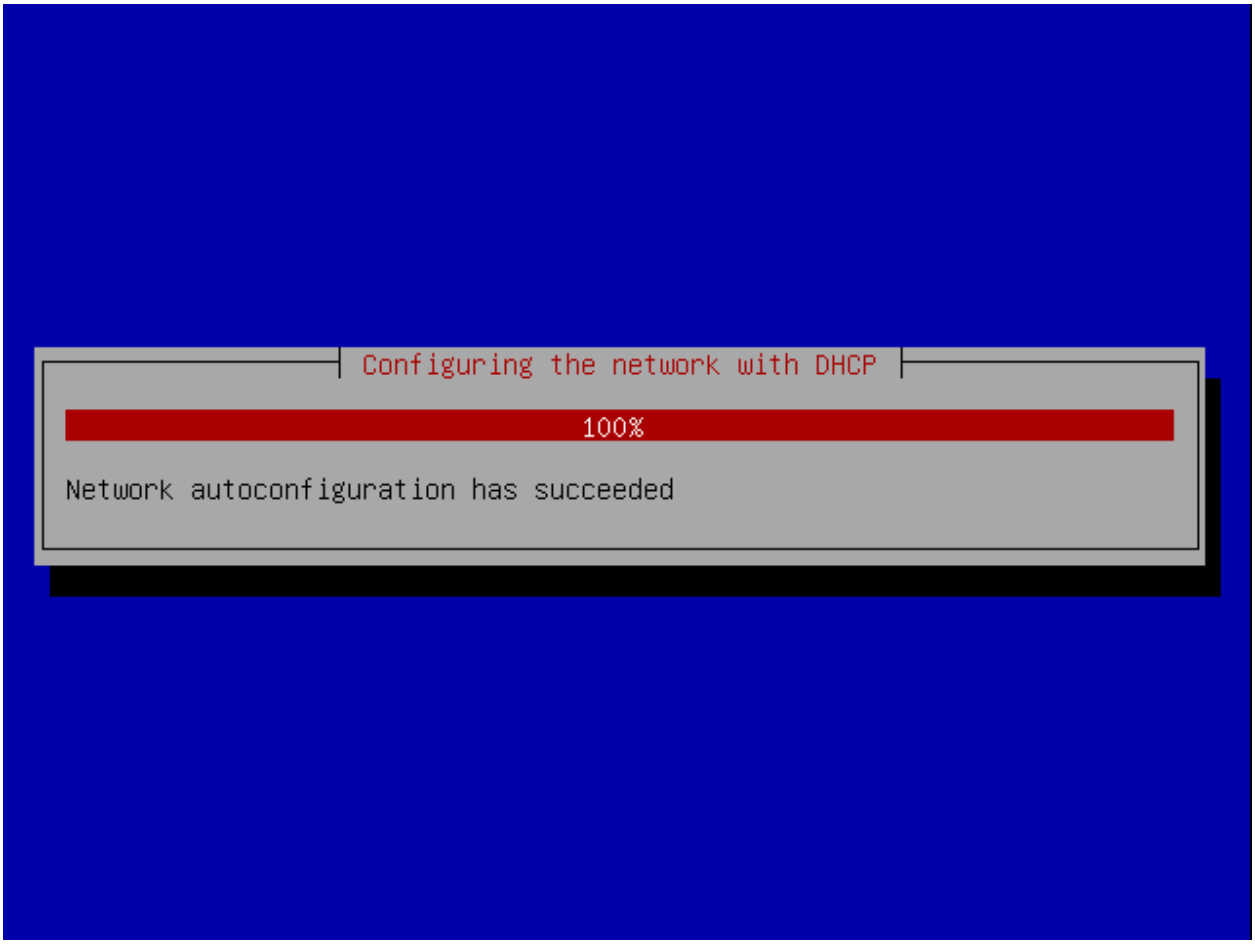

OpenMediaVault
The open network attached storage solution
Boot menu
Install
Install (serial console)
Help
Automatic boot in 7 seconds...

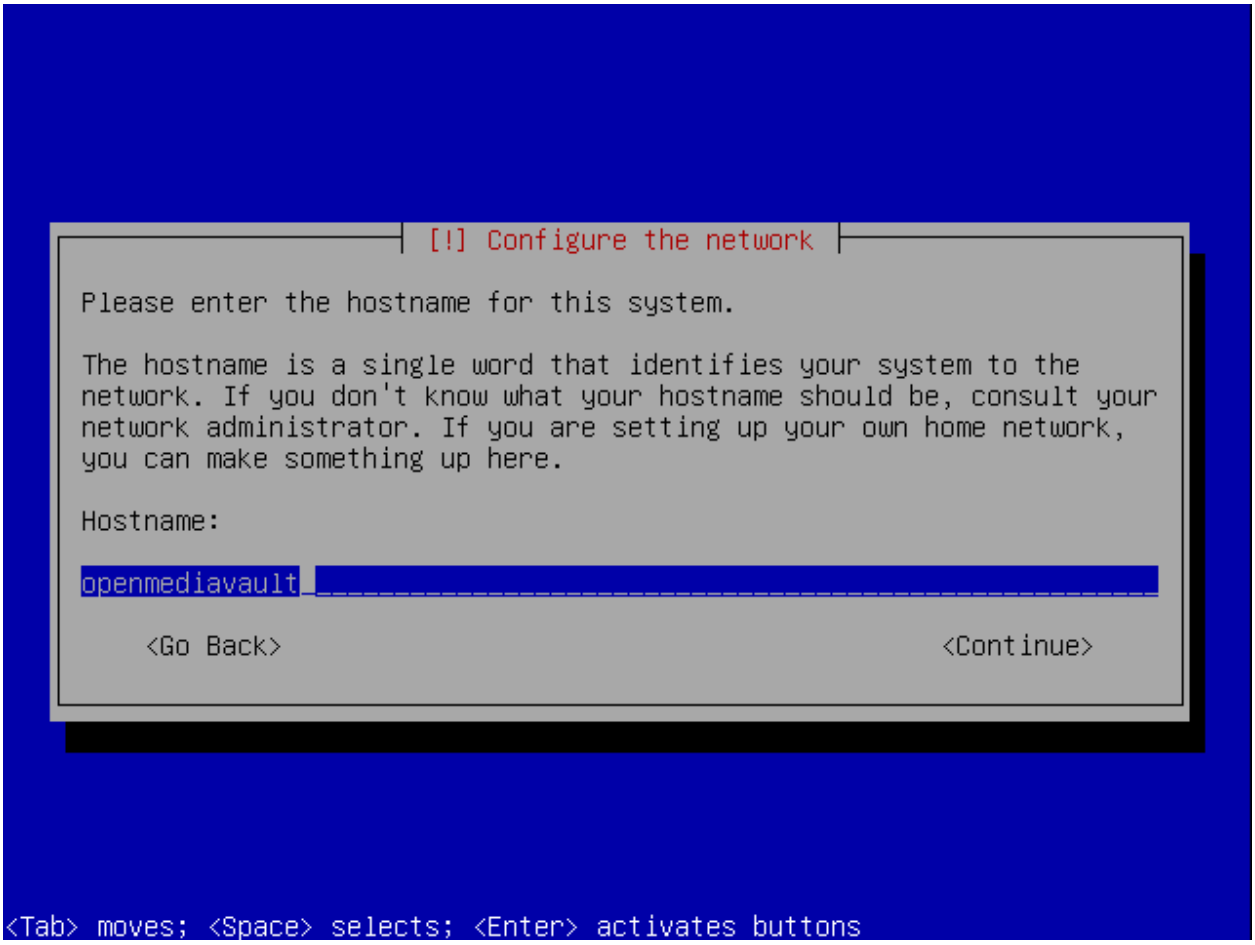

[!!] Select a language
Choose the language to be used for the installation process. The selected language will also be the default language for the installed system.
Language:
Albanian - Shqip
Arabic - عربي
Asturian - Asturianu
Basque - Euskara
Belarusian - Беларуская
Bosnian - Bosanski
Bulgarian - Български
Catalan - Català
Chinese (Simplified) - 中文(简体)
Chinese (Traditional) - 中文(繁體)
Croatian - Hrvatski
Czech - Čeština
Danish - Dansk
Dutch - Nederlands
English - English
<Go Back>
<Tab> moves; <Space> selects; <Enter> activates buttons

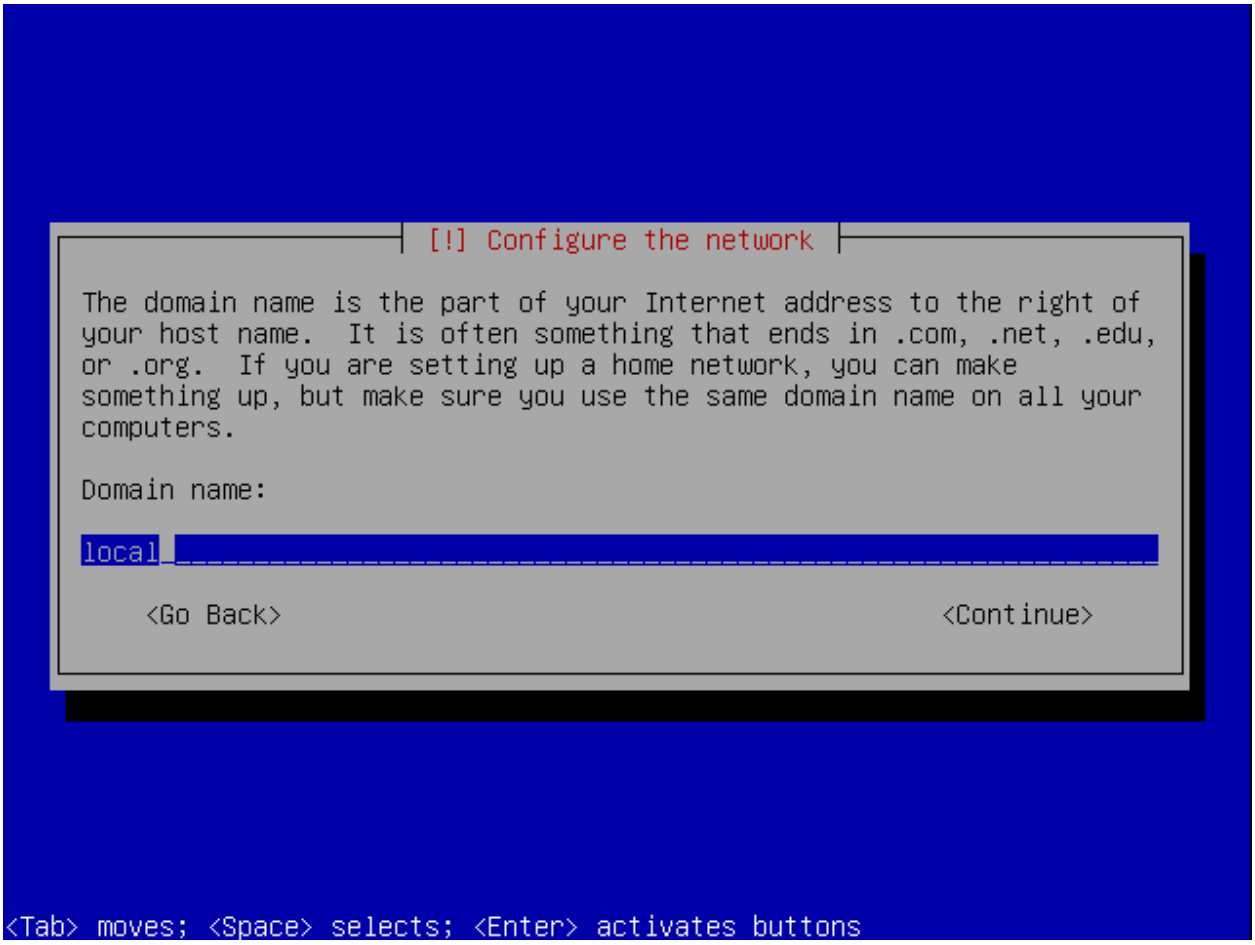

[!!] Select your location
The selected location will be used to set your time zone and also for example to help select the system locale. Normally this should be the country where you live.
This is a shortlist of locations based on the language you selected. Choose "other" if your location is not listed.
Country, territory or area:
Botswana
Canada
Hong Kong
India
Ireland
New Zealand
Nigeria
Philippines
Singapore
South Africa
United Kingdom
United States
<Go Back>
<Tab> moves; <Space> selects; <Enter> activates buttons

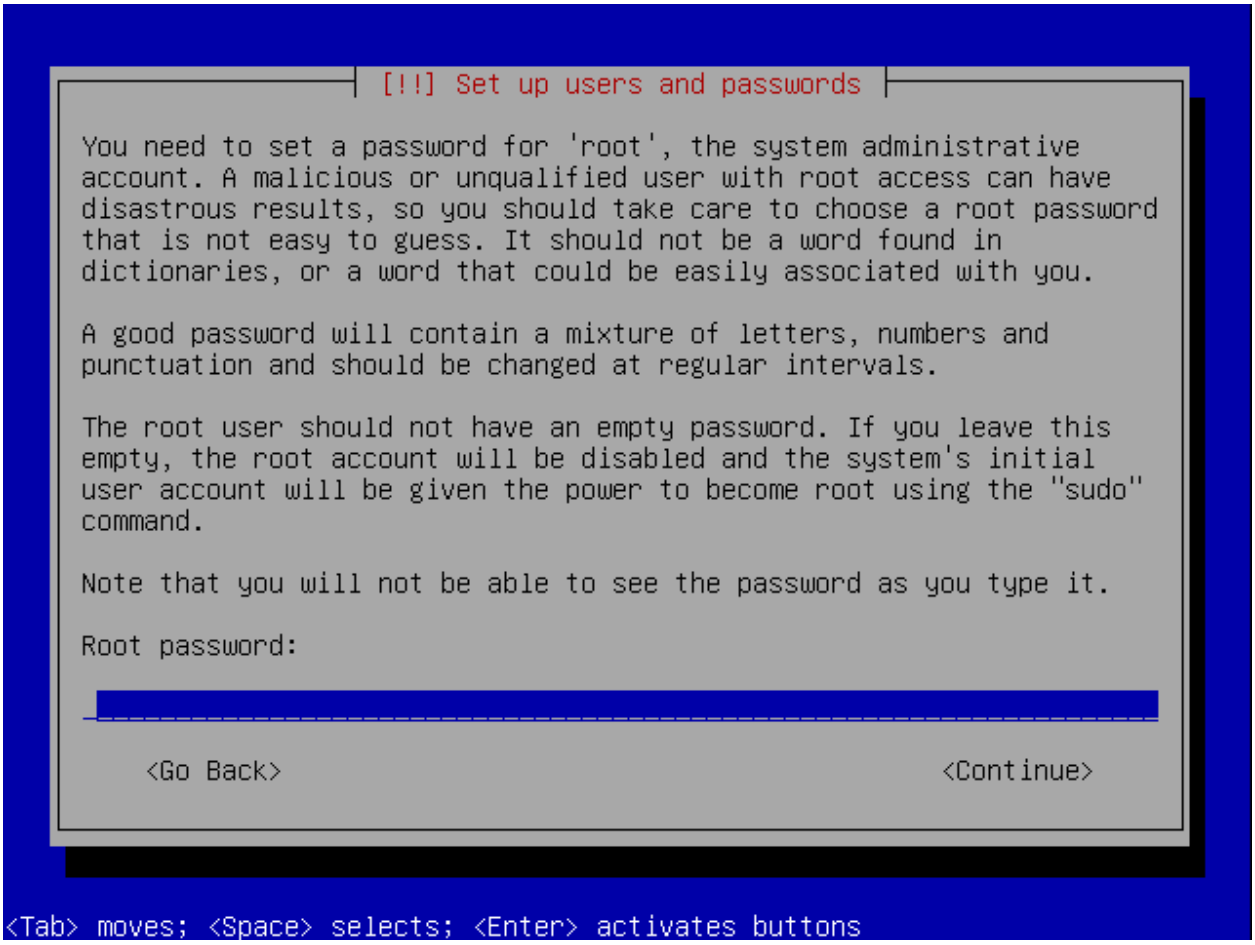

[!] Select a keyboard layout
Keymap to use:
American English
Belarusian
Belgian
Brazilian (ABNT2 layout)
Brazilian (EUA layout)
British English
Bulgarian
Canadian French
Canadian Multilingual
Croatian
Czech
Danish
Dutch
Dvorak
Estonian
Finnish
French
German
Greek
<Go Back>
<Tab> moves; <Space> selects; <Enter> activates buttons

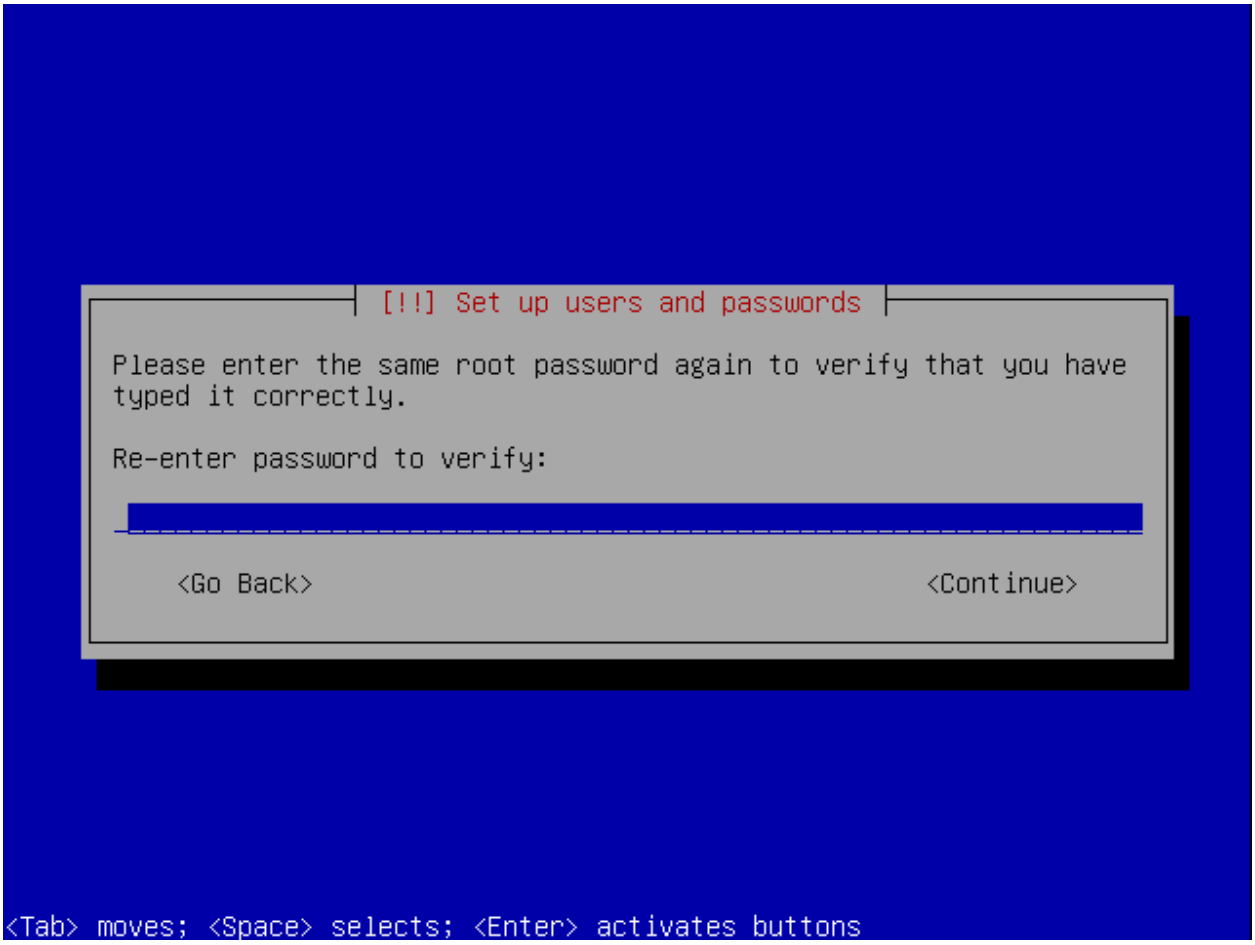

Loading additional components
37%
Retrieving partman-ext3

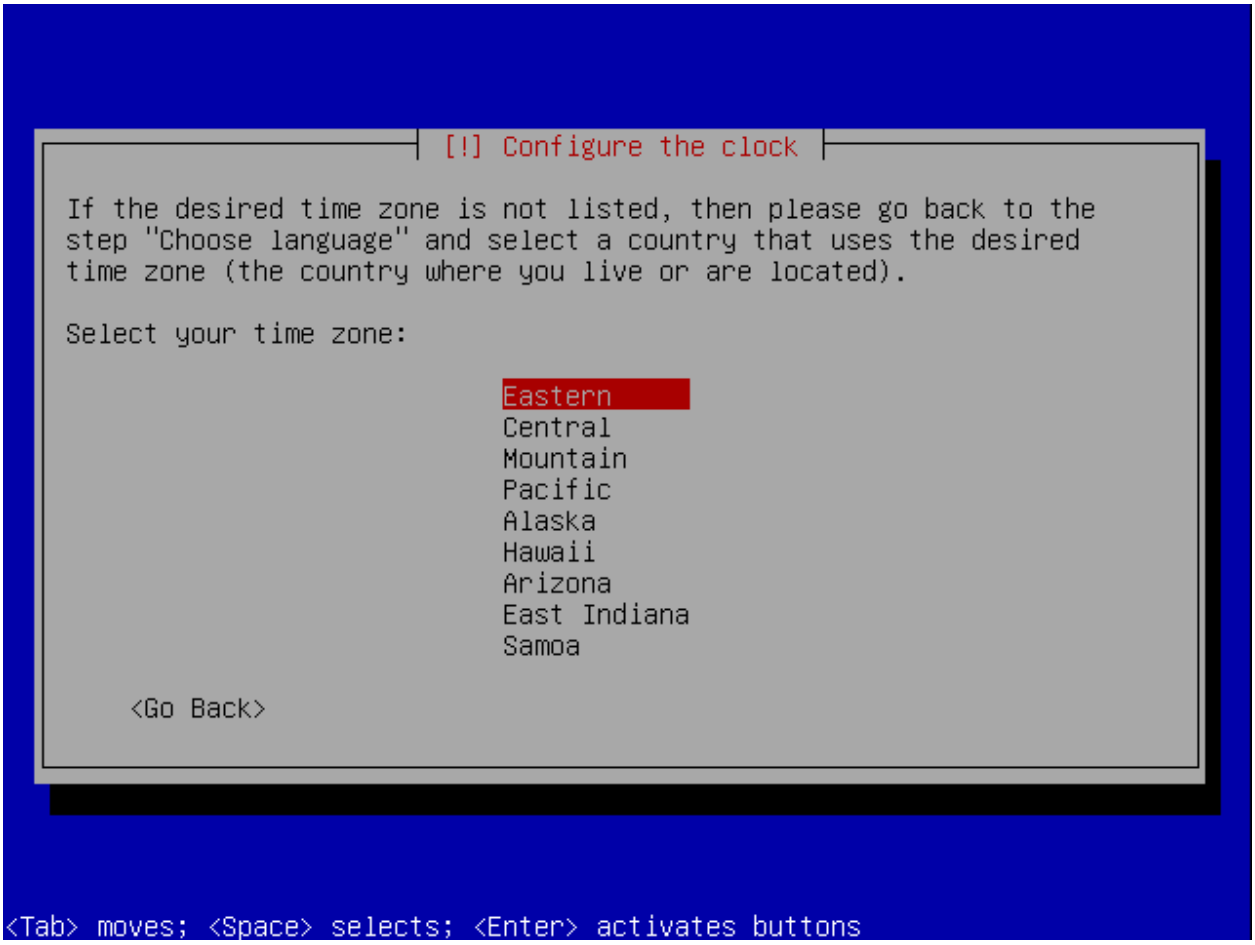

Configuring the network with DHCP
100%
Network autoconfiguration has succeeded

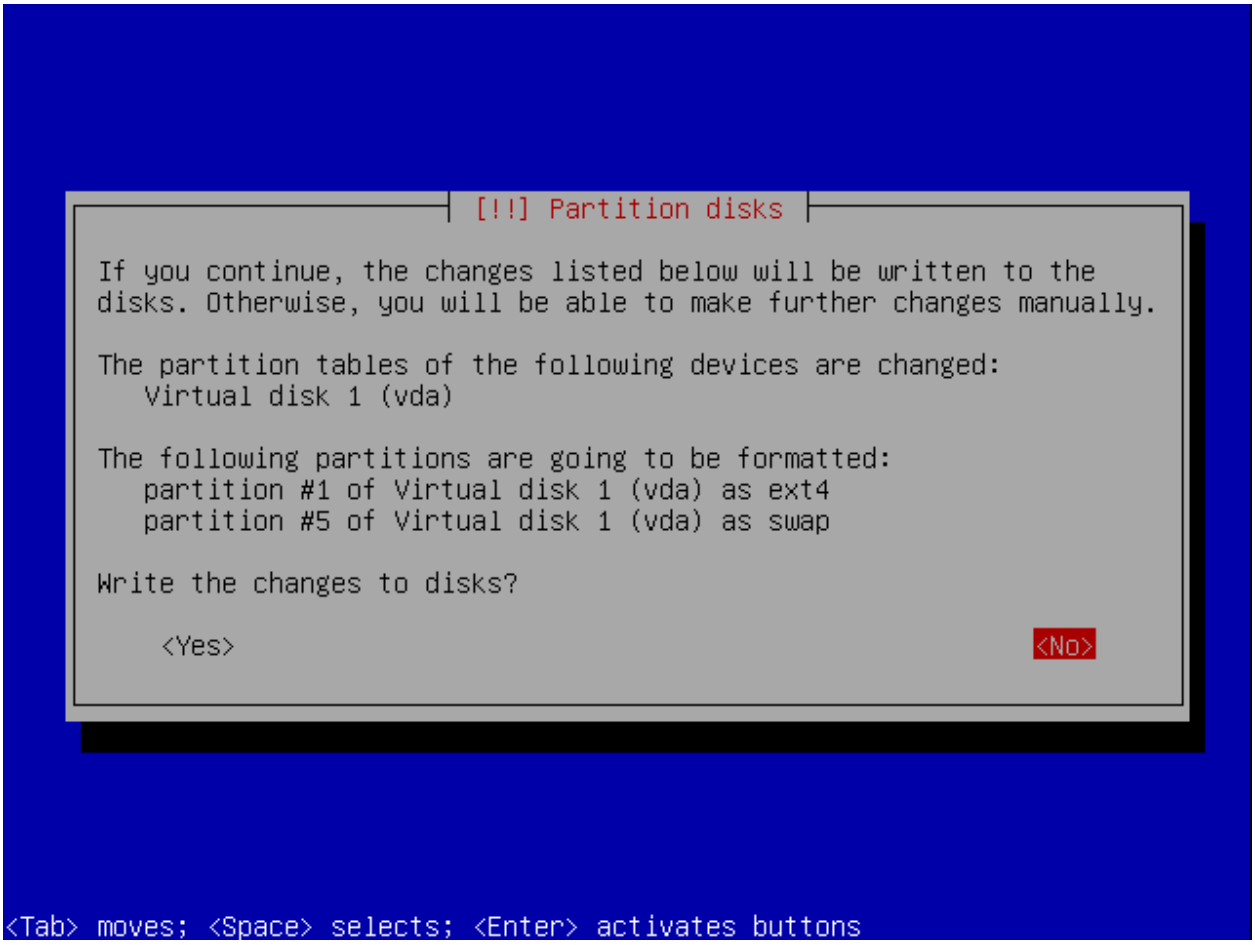

[!] Configure the network
Please enter the hostname for this system.
The hostname is a single word that identifies your system to the network. If you don't know what your hostname should be, consult your network administrator. If you are setting up your own home network, you can make something up here.
Hostname:
openmediavault
<Go Back>
<Continue>
<Tab> moves; <Space> selects; <Enter> activates buttons

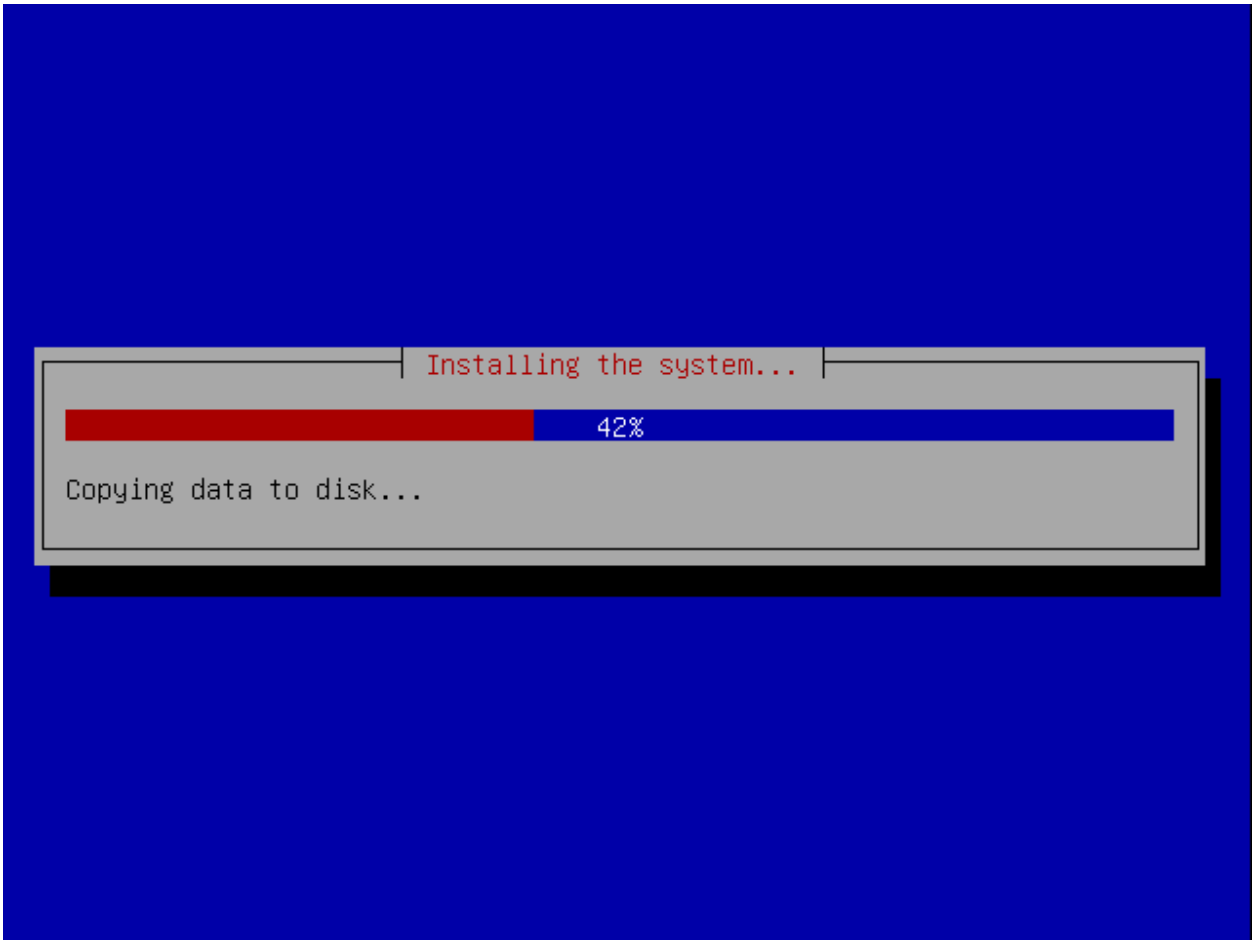

[!] Configure the network
The domain name is the part of your Internet address to the right of your host name. It is often something that ends in .com, .net, .edu, or .org. If you are setting up a home network, you can make something up, but make sure you use the same domain name on all your computers.
Domain name:
local
<Go Back>
<Continue>
<Tab> moves; <Space> selects; <Enter> activates buttons

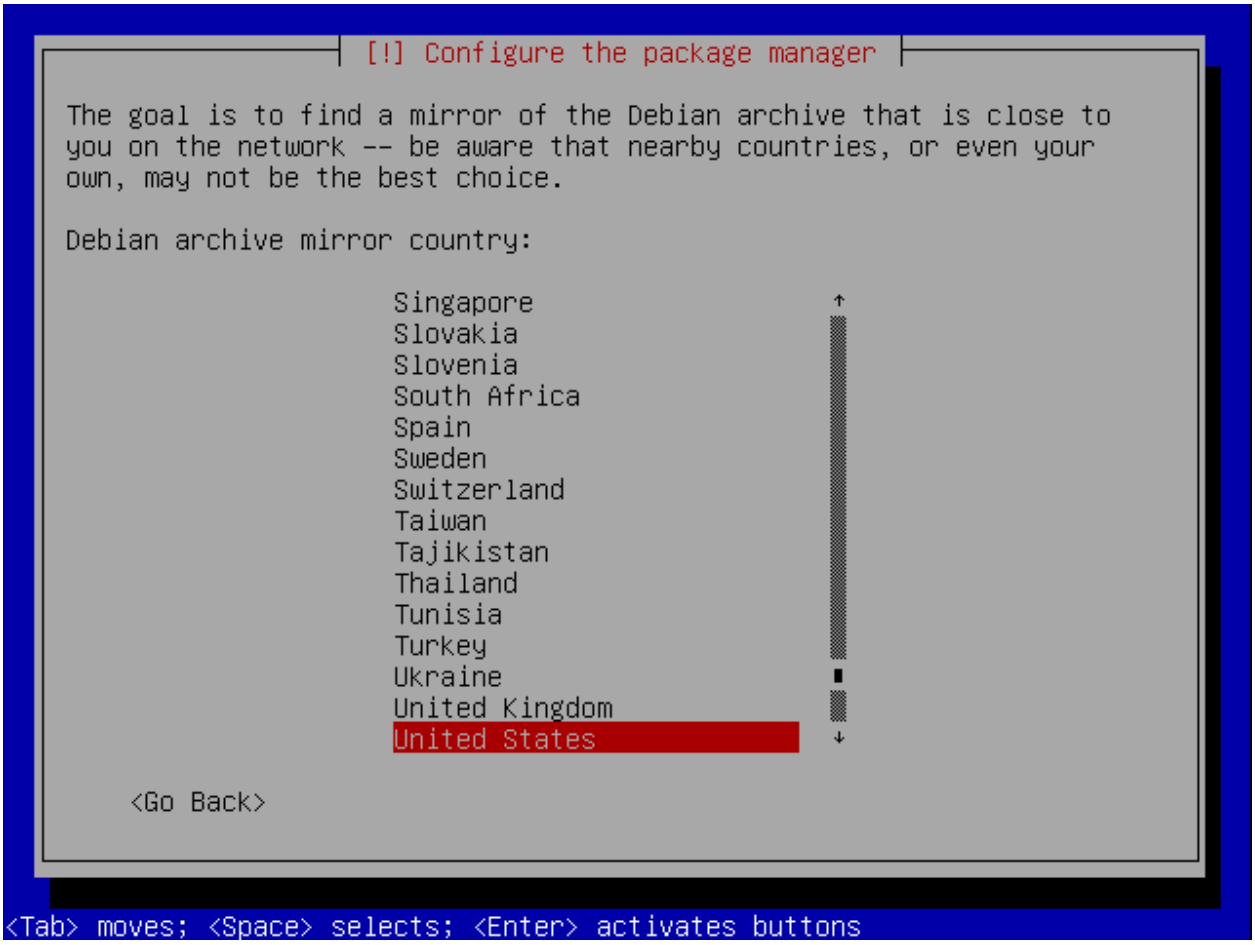

[!!] Set up users and passwords
You need to set a password for 'root', the system administrative account. A malicious or unqualified user with root access can have disastrous results, so you should take care to choose a root password that is not easy to guess. It should not be a word found in dictionaries, or a word that could be easily associated with you.
A good password will contain a mixture of letters, numbers and punctuation and should be changed at regular intervals.
The root user should not have an empty password. If you leave this empty, the root account will be disabled and the system's initial user account will be given the power to become root using the "sudo" command.
Note that you will not be able to see the password as you type it.
Root password:
<Go Back>
<Continue>
<Tab> moves; <Space> selects; <Enter> activates buttons

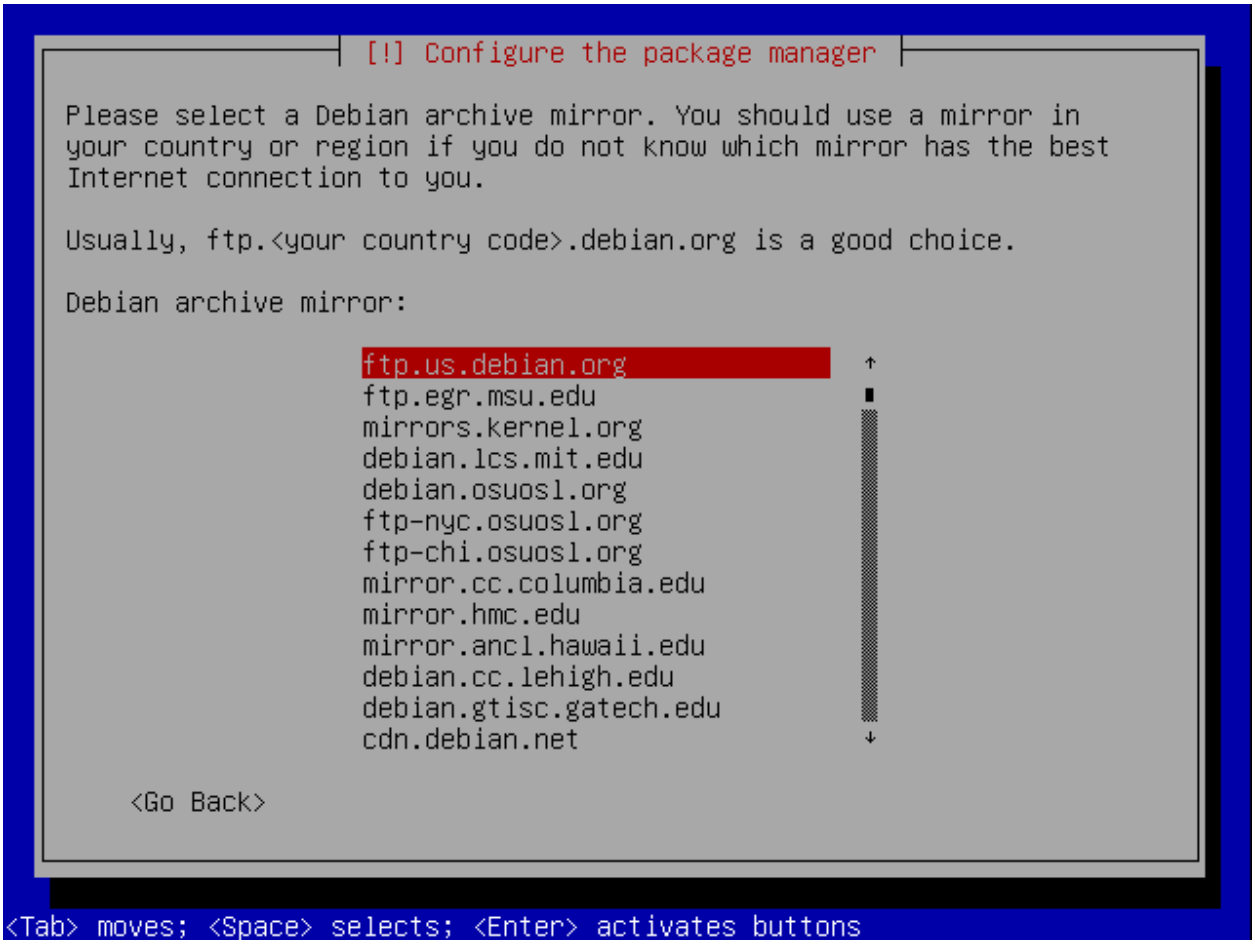

[!!] Set up users and passwords
Please enter the same root password again to verify that you have typed it correctly.
Re-enter password to verify:
<Go Back>
<Continue>
<Tab> moves; <Space> selects; <Enter> activates buttons

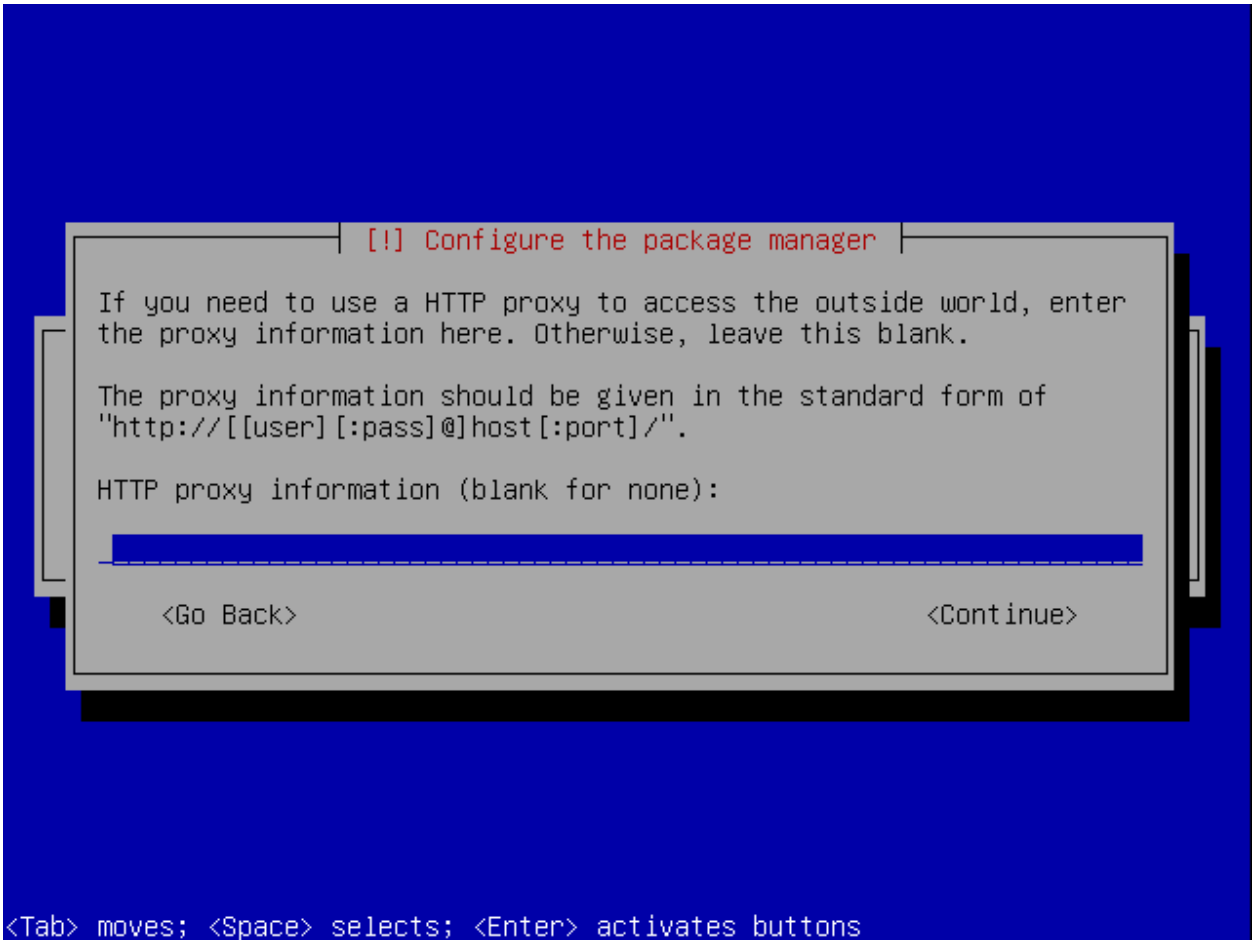

[!] Configure the clock
If the desired time zone is not listed, then please go back to the step "Choose language" and select a country that uses the desired time zone (the country where you live or are located).
Select your time zone:
Eastern
Central
Mountain
Pacific
Alaska
Hawaii
Arizona
East Indiana
Samoa
<Go Back>
<Tab> moves; <Space> selects; <Enter> activates buttons

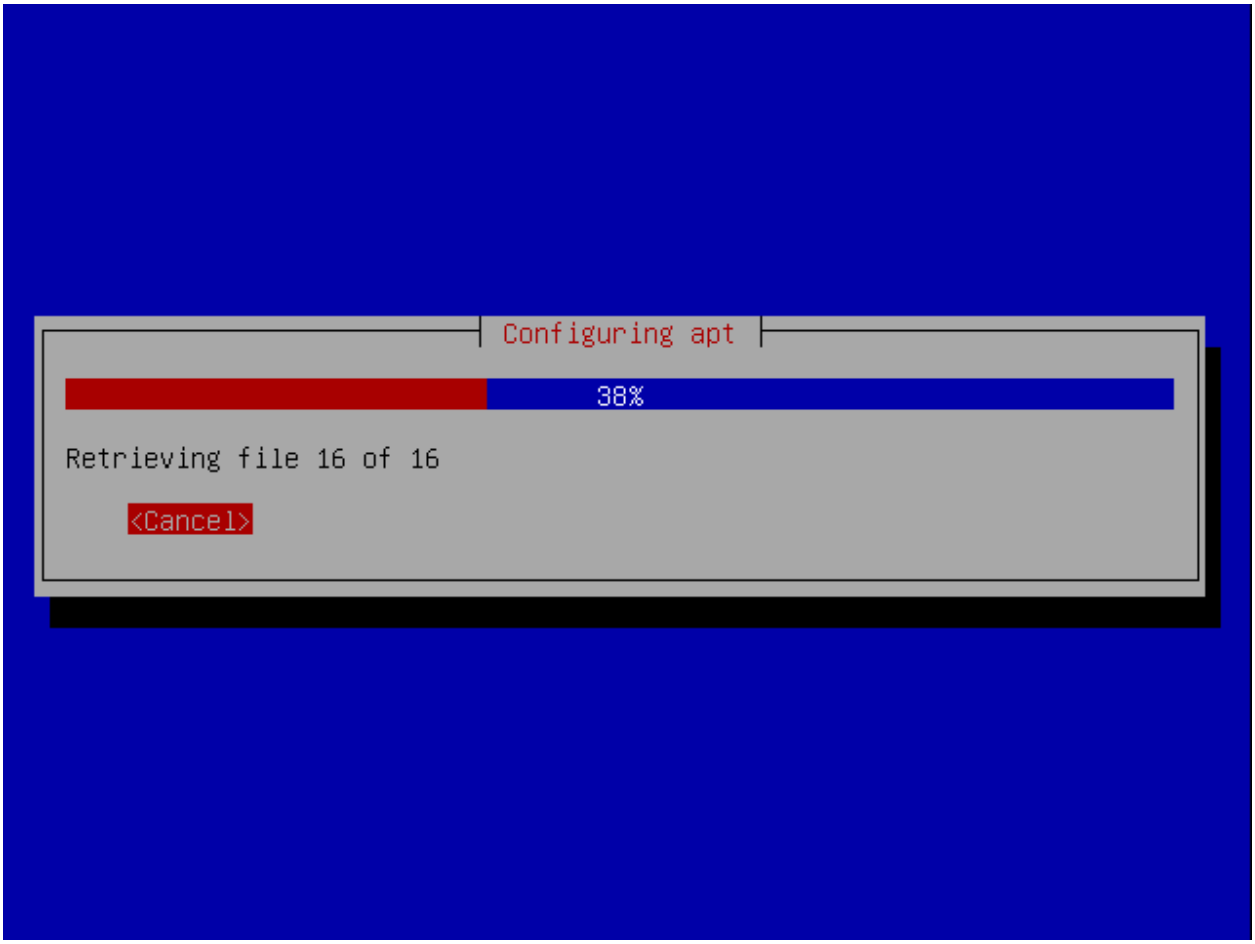

[!!] Partition disks
If you continue, the changes listed below will be written to the disks. Otherwise, you will be able to make further changes manually.
The partition tables of the following devices are changed:
Virtual disk 1 (vda)
The following partitions are going to be formatted:
partition #1 of Virtual disk 1 (vda) as ext4
partition #5 of Virtual disk 1 (vda) as swap
Write the changes to disks?
<Yes>
<No>
<Tab> moves; <Space> selects; <Enter> activates buttons

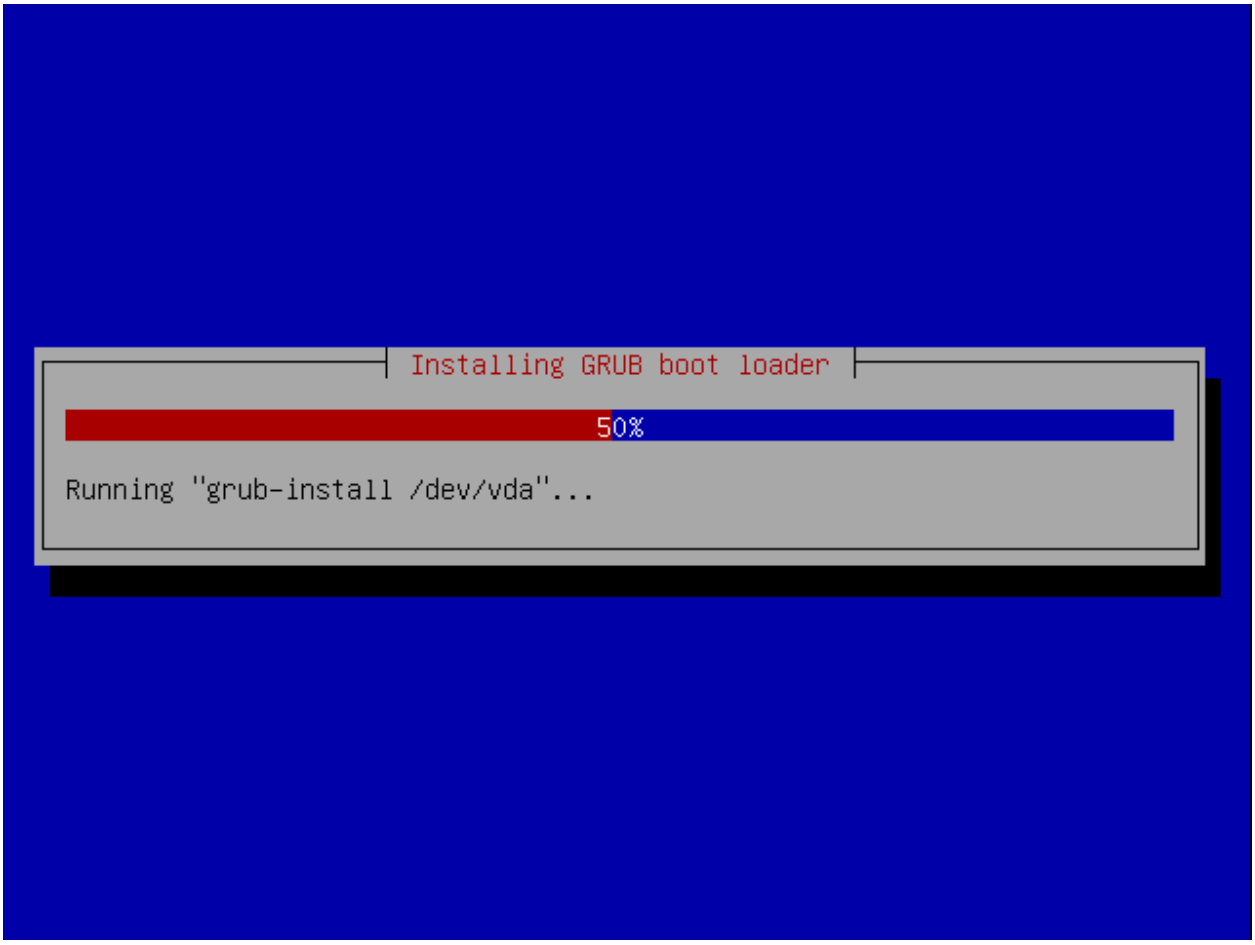

Installing the system...
42%
Copying data to disk...

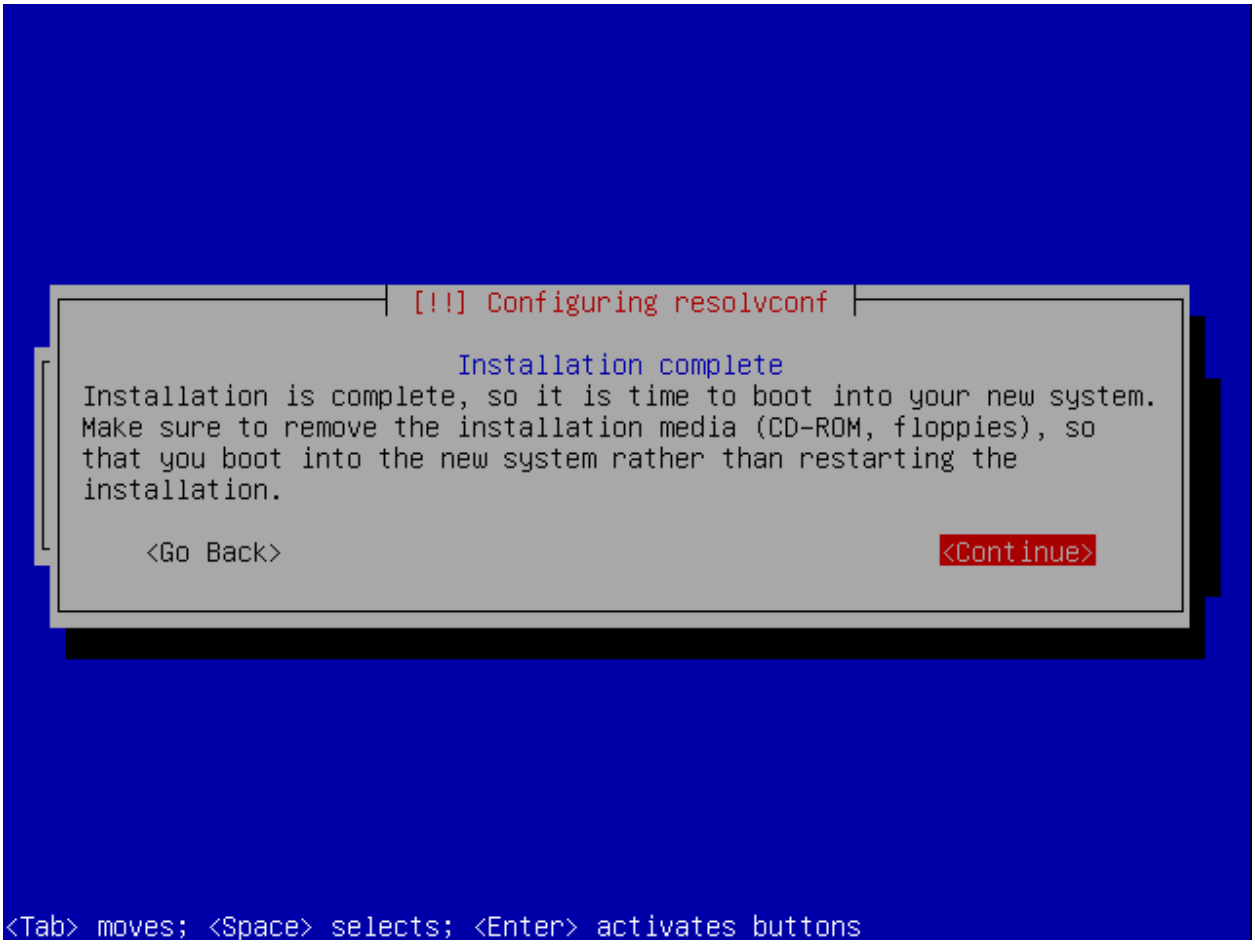

[!] Configure the package manager
The goal is to find a mirror of the Debian archive that is close to you on the network -- be aware that nearby countries, or even your own, may not be the best choice.
Debian archive mirror country:
Singapore
Slovakia
Slovenia
South Africa
Spain
Sweden
Switzerland
Taiwan
Tajikistan
Thailand
Tunisia
Turkey
Ukraine
United Kingdom
United States
<Go Back>
<Tab> moves; <Space> selects; <Enter> activates buttons


[!] Configure the package manager
Please select a Debian archive mirror. You should use a mirror in
your country or region if you do not know which mirror has the best
Internet connection to you.
Usually, ftp.<your country code>.debian.org is a good choice.
Debian archive mirror:
ftp.us.debian.org
ftp.egr.msu.edu
mirrors.kernel.org
debian.lcs.mit.edu
debian.osuosl.org
ftp-nyc.osuosl.org
ftp-chi.osuosl.org
mirror.cc.columbia.edu
mirror.hmc.edu
mirror.ancl.hawaii.edu
debian.cc.lehigh.edu
debian.gtisc.gatech.edu
cdn.debian.net
<Go Back>
<Tab> moves; <Space> selects; <Enter> activates buttons


[!] Configure the package manager
If you need to use a HTTP proxy to access the outside world, enter
the proxy information here. Otherwise, leave this blank.
The proxy information should be given in the standard form of
"http://[[user][:pass]@]host[:port]/".
HTTP proxy information (blank for none):
<Go Back>
<Continue>
<Tab> moves; <Space> selects; <Enter> activates buttons


Configuring apt
38%
Retrieving file 16 of 16
<Cancel>


Installing GRUB boot loader
50%
Running "grub-install /dev/vda"...


[!!] Configuring resolvconf
Installation complete
Installation is complete, so it is time to boot into your new system.
Make sure to remove the installation media (CD-ROM, floppies), so
that you boot into the new system rather than restarting the
installation.
<Go Back>
<Continue>
<Tab> moves; <Space> selects; <Enter> activates buttons

### 3.1.5 Troubleshooting

**The error *Unable to install GRUB in /dev/sda* occurs.**

In this case execute the following steps:

- Select *Continue* in this window and also on the next which says *Installation step failed*.
- In the Debian installer main menu (which should have popped up by now), select *Execute a shell* and then *Continue*.
- Execute the following commands:

```
# Chroot.
chroot /target
# Replace [a-z] with the drive you want to install grub to.
# This is normally the drive you've selected to install OpenMediaVault on.
grub-install /dev/sd[a-z]
# Update GRUB.
update-grub
# Exit chroot.
exit
# Exit shell.
exit
```

- Select *Continue without boot loader* in the Debian installer main menu and then *Continue*.
- It should now continue the installation successfully.

## 3.2 Installation on USB

openmediavault can be installed on an USB flash drive to boot from it.

1. Follow the *Installation via ISO image* and select your USB flash drive as target drive.
2. Enable the openmediavault-writecache plugin. This lowers the amount of writes to the USB flash drive, making sure you can enjoy it for a long time.

## 3.3 Installation on Debian

You can install openmediavault on an existing Debian 13 installation as well.

**Important**

The installation of openmediavault will be denied if a graphical desktop environment is detected.

**Note**

openmediavault does not import any existing settings of your system, except for network and time related settings. Existing settings will be overwritten if the service is managed by openmediavault. Please reconfigure these services via the web interface.

**Important**

Ensure that you have a working Internet connection during installation. An offline installation will lead to problems later on.

To do so, simply install the system using the Debian netinst images. After that apply the commands below. Please do **not** install a graphical desktop environment or web server, use a minimal server installation only (SSH server and standard system utilities). For a step by step install guide have a look into the Debian minimal install guide.

On ARM devices, please check if there's an appropriate Armbian image available. Make sure you are using the correct *Debian version* that openmediavault is based on. After installing Armbian, then use the *armbian-config* tool to install openmediavault in a single step with all performance and reliability tweaks included. If there's no Armbian image for your device, simply follow the steps outlined below.

On Raspberry Pi OS the below instructions only partially work. Please refer to a specific installation script.

**Warning**

Before installing the openmediavault package as described below, ensure that the *systemd-resolved* package is already installed and configured on the system. Otherwise the package will be installed automatically with openmediavault, which will inevitably result in DNS resolution no longer functioning. This in turn means that the openmediavault package cannot be installed successfully, as it downloads and installs additional packages during installation which requires functioning network connectivity.

**Note**

The following commands must be executed as `root` user.

Install and configure the *systemd-resolved* if not already installed. The configuration is only temporary and gets lost after a reboot. After the installation of the openmediavault package, this can be made permanently by reconfiguring the network using the **`omv-firstaid`** command.

```
apt-get install --yes systemd-resolved psmisc
systemctl enable --now systemd-resolved.service
systemctl restart systemd-resolved.service
resolvectl dns <INTERFACE> <DNS_SERVER_IP>
```

**Note**

If your IP address is configured by DHCP, the dhcp client may interfere with *systemd-resolved*, preventing the download of additional packages (`apt-get` and `wget` fails). In that case, stop it with `killall dhcpcd` and repeat the `resolvectl dns <INTERFACE> <DNS_SERVER_IP>` command.

Install the openmediavault keyring manually:

```
apt-get install --yes gnupg
wget --quiet --output-document=- https://packages.openmediavault.io/archive.key | gpg --
↪dearmor --yes --output "/usr/share/keyrings/openmediavault-archive-keyring.gpg"
```

Add the openmediavault package repositories:

```
cat <<EOF >> /etc/apt/sources.list.d/openmediavault.list
deb [signed-by=/usr/share/keyrings/openmediavault-archive-keyring.gpg] http://packages.
↪openmediavault.org/public/ synchrony main
deb [signed-by=/usr/share/keyrings/openmediavault-archive-keyring.gpg] http://packages.
↪openmediavault.io/ synchrony main
# deb [signed-by=/usr/share/keyrings/openmediavault-archive-keyring.gpg] http://
↪downloads.sourceforge.net/project/openmediavault/packages/ synchrony main
## Uncomment the following line to add software from the proposed repository.
# deb [signed-by=/usr/share/keyrings/openmediavault-archive-keyring.gpg] http://packages.
↪openmediavault.org/public/ synchrony-proposed main
# deb [signed-by=/usr/share/keyrings/openmediavault-archive-keyring.gpg] http://packages.
↪openmediavault.io/ synchrony-proposed main
# deb [signed-by=/usr/share/keyrings/openmediavault-archive-keyring.gpg] http://
↪downloads.sourceforge.net/project/openmediavault/packages/ synchrony-proposed main
## This software is not part of OpenMediaVault, but is offered by third-party
## developers as a service to OpenMediaVault users.
# deb [signed-by=/usr/share/keyrings/openmediavault-archive-keyring.gpg] http://packages.
↪openmediavault.org/public/ synchrony partner
# deb [signed-by=/usr/share/keyrings/openmediavault-archive-keyring.gpg] http://packages.
↪openmediavault.io/ synchrony partner
# deb [signed-by=/usr/share/keyrings/openmediavault-archive-keyring.gpg] http://
↪downloads.sourceforge.net/project/openmediavault/packages/ synchrony partner
EOF
```

**Note**

If you are a user in mainland China, TUNA provides mirroring services.

Install the openmediavault package:

```
export LANG=C.UTF-8
export DEBIAN_FRONTEND=noninteractive
export APT_LISTCHANGES_FRONTEND=none
apt-get update
apt-get --yes --auto-remove --show-upgraded \
    --allow-downgrades --allow-change-held-packages \
    --no-install-recommends \
    --option DPkg::Options::="--force-confdef" \
    --option DPkg::Options::="--force-confold" \
    install openmediavault
```

Populate the openmediavault database with several existing system settings, e.g. the network configuration:

```
omv-confdbadm populate
```

**Note**

Right now only `/etc/network/interfaces` is parsed to get the current network configuration. If the network is configured a different way (e.g. via *systemd* or *NetworkManager*), then the database is not populated and does not contain the necessary information to deploy the network configuration with *netplan* for *systemd-networkd* and *systemd-resolved*. In that case use **`omv-firstaid`** to do the initial network configuration instead of the following

step.

Re-deploy the network configuration via the services used by openmediavault:

```
omv-salt deploy run systemd-networkd
```

Or alternatively use **omv-firstaid** to do the initial network configuration from CLI.

> **Note**
>
> The IP address may change during the redeployment of the network configuration, therefore you may lose the connection when you are connected via SSH.

By default the *root* user can now access the system via SSH as a fallback if something went wrong during the installation, e.g. the UI is not accessible. SSH access for *root* should be disabled for security reasons as soon as possible after the installation has been successfully finished.

> **Note**
>
> The user created by the Debian installer will not able to SSH into the system after openmediavault has been installed. This is because only users who are assigned to the *_ssh* group are allowed to use SSH.

## 3.4 Installation via images

### 3.4.1 Prepare the image

Etcher can also be used to transfer ARM images into SD cards.

### 3.4.2 Boot the installer

ARM images are ready to go, so boot from SD card and wait for the initial setup to complete. This takes around 30 minutes on slow devices like Raspberry PI.

> **Warning**
>
> Do not shut down the system during the initial setup. Give the system enough time to finish the setup stage, otherwise the system will be in a broken state.

CHAPTER

FOUR

# FEATURES

## 4.1 System

## 4.2 General settings

**General settings:** Change web interface listening port, SSL and force SSL. Change admin password.

**Notification system:** Integrated into several services in the form of email using Postfix[1] backend as MTA, these include scheduled tasks, services monitoring, S.M.A.R.T., MDADM and cron-apt. Since openmediavault 3.0 it is possible to add also third party notification systems by using scripts, more information can be found here and real example on how to use it can be found here.

**Network configuration:** The web interface provides configuration options for ethernet, WiFi (only WPA/WPA2 supported), bond and vlan interfaces. This also includes a panel for firewall configuration.

**Certificates:** Create or import existing SSL and SSH certificates. This certificates can be used for securing the web interface or SSH access. Plugins can use the backend framework to select the available certificates. Automatic scheduled tasks that check for expired SSL certificates, including notification by email.

**Power Management:** Scheduled power management for hibernation (S5), suspend (S3), shutdown and/or reboot.

**Service Discovery:** Using avahi-daemon[2] is possible to announce the following services Samba, NFS, AFP, FTP, web admin panel, to any Linux desktop with file browser that supports it (GNOME, KDE or XFCE for example). OS X can recognise AFP and Samba services in the Finder sidebar. To announce SMB to windows clients, samba uses NetBios, not avahi.

**Scheduled Tasks:** Based on cron the webUI can define tasks for running specific commands or custom scripts at certain time or regular intervals.

**Update Manager:** Displays all available packages for upgrade. Automatic daily tasks to check for updates, including notification by email. Security updates will be installed unattended.

## 4.3 Storage

**S.M.A.R.T.:** Based on smartmontools[3], It can display advanced information of S.M.A.R.T values in the webUI. It can also schedule health tests as well as send notifications when S.M.A.R.T. attributes values change.

**RAID Management:** Based Linux RAID[4], create arrays in 6 different configurations. Levels available are linear, 0, 1, 10, 5 and 6. The array can have disks removed or expanded using the web interface.

**File Systems:** Volume format, device mount and unmount. More information in the *filesystem section*.

[1] https://www.postfix.org/
[2] https://www.avahi.org/
[3] https://www.smartmontools.org/
[4] https://raid.wiki.kernel.org/index.php/RAID_setup

**LVM:** Enhanced by the LVM2 plugin, the system has the capability of formatting disks or partitions as LVM that can be used in volume groups to create logical partitions.

**Shared Folders:** Simple shared folder administration. Within this section is also possible to assign ACLs and/or privileges to the shared folders. Snapshots can be taken manually or via scheduled tasks for shared folders that are located on Btrfs file systems. Automatic scheduled tasks for Btrfs file systems to scrub them and check for errors, including notifications via email.

## 4.4 Users

**Users:** User and group managing. Using privileges is possible to restrict access/login to shares on network sharing services (FTP, Samba and AFP) without interfering Unix permissions. Automatic scheduled task that checks for locked/banned users, including notification by email.

**Groups:** Create and manage custom groups. System groups cannot be manipulated here.

## 4.5 Services

**SMB/CIFS:** SMB sharing protocol using Samba[5] as standalone server by default.

**FTP:** Service based on proftpd[6]. Intended for accessing shares from remote or local.

> **Note**
>
> This service has been removed from core in openmediavault version 6. It can be installed as plugin now.

**RSync:** Server daemon. Shared folders can be defined as rsyncd modules. With scheduled tasks, rsync client can be configured for push or pull jobs.

**NFS:** Network file system protocol[7].

**SSH:** Remote shell access using openssh[8].

**TFTP:** A basic configuration panel is provided. This can complement a PXE server to deploy local network installations.

> **Note**
>
> This service has been removed from core in openmediavault version 4. It can be installed as plugin now.

## 4.6 Diagnostics

**Dashboard:** By default the server comes with four information widgets. Network interfaces, System, Filesystem and service/daemon status. The dashboard panel can have widgets added using the plugin framework.

**System information:** The panel displays four tabs with system information and statistics graphs.

**System Logs:** Interface to view and download logs from syslog, journalctl, message, auth, ftp, rsync and samba. Plugins can attach their logs here using the framework.

[5] https://www.samba.org/
[6] http://www.proftpd.org/
[7] https://nfs.sourceforge.net/
[8] https://www.openssh.com/

**Services:** View status (enabled/disabled and running/not running) of services. Detailed information is provided by default for Samba, FTP and SSH. Plugins can use this tab to integrate their service information also.

CHAPTER

FIVE

# ADMINISTRATION

Configuration changes are not applied immediately, instead you can determine when this happens. This enables several changes to be activated at the same time, which reduces unnecessary waiting times.

> **Note**
>
> openmediavault does not display the submodules that are affected by the configuration changes. If you still want to know which submodules are affected, simply run `cat /var/lib/openmediavault/dirtymodules.json` in the CLI.

## 5.1 General

### 5.1.1 Workbench

Here you can make settings that affect the web interface of openmediavault.

For example, you can change the port at which the web interface can be reached. You can also activate or force HTTPS.

By default, the user is automatically logged out of the web interface after a certain period of inactivity. You can adjust this time or deactivate the functionality here.

> **Note**
>
> If you make changes here and activate them, error messages may appear in the web interface because the web server is restarted in the background. These error messages can be ignored.

### 5.1.2 Network

In this section you can set several system network related settings.

#### General

Hostname and domain settings. If your system is part of a private network without a registered domain, use *internal* as suggested by ICANN.

#### Interfaces

Only network interfaces that have been configured via the web interface are displayed in the data table. If this panel is empty right after the installation the automatic synchronisation of the interface configuration that was done by the installer. In this case simply start to configure the interfaces via the web interface.

The dashboard contains a network widget that displays the current status of the interfaces.

### Ethernet

Just select DHCP or static. openmediavault is a server so the recommended setting is to have static IP address, if you have a proper network infrastructure (separate router and switch). In a reboot, if the router fails to boot you can still access the web interface through the switch bridge. If the switch also fails you can use a direct Ethernet connection with the static IP address assigned to the server NIC.

Do not leave DNS setting empty; it is essential for fetching updates. A common value is to use the same IP address as the gateway. If unsure, just use google DNS `8.8.8.8`.

**Wake on LAN (WOL)**
: This enables WOL in the kernel driver, make sure the NICs supports this and the feature is enabled in BIOS.

### Wireless

The configuration window displays the same IP configuration fields as Ethernet, plus the relevant wireless values: SSID (the wireless network name) and the password.

Whenever possible, use Ethernet for a NAS server. Wireless should not be used in a production server; this feature is intended for extreme cases only.

### VLAN

If your network supports VLAN, just add the parent interface and the VLAN id.

### Bond

The configuration window provides all available modes for the bond driver. To configure bonding, is necessary at least two physical network interfaces. The web interface allows the selection of less than two. This is by design for configuration purposes. The workflow is as follow for dual NICs:

- If the primary NIC is already working either by the installer, configure it through the web interface as static. If set as static using the same IP address given by DHCP, it should not be necessary to re-login to the web interface.
- Click `Network | Interfaces | Add | Bond`, select the second available NIC, select the bond mode, fill the IP field and subnet mask values, leave gateway and DNS empty. Save and hit apply.
- Log out and access the web interface using the new IP address assigned to the bond interface created.
- Now select the primary interface configured through web interface in the first step, and delete it. Save and hit apply.
- Select the newly created bond interface, click edit add now the physical nic that was deleted from the step before should be available to select. Save and hit apply.
- The dashboard should now report the bond interface information (including speed).

**Note**

- 802.3ad LACP (Link Aggregation) mode only works if physical interfaces are connected to a managed switch that supports aggregation.
- Is not possible to achieve 2GBit bandwidth (or more depending on the number of NICs) in a single client using LACP, even if the client also has a LACP-bonded NIC or 10Gbit card; there is no multipath support in Samba or other openmediavault services in the way Windows Server has for file sharing using SMB.
- Higher speeds using link aggregation are limited by disk speed. When serving simultaneous clients make sure the physical media is capable of reaching the speed of the bonded NIC (e.g. SSD or RAID array).

### Advanced Interface Configuration

#### Proxy

This panel configures the server proxies using system wide environmental variables. All software that obeys Linux proxy environmental variables should be able to use the proxy. This is useful for example if there are many Debian servers in the network, when performing **apt** operations, packages can be cached in the proxy if this configured appropriately to reduce download bandwidth.

The variables name are:

```
http_proxy
https_proxy
ftp_proxy
```

This settings do not configure openmediavault to act as a proxy server.

#### Service Discovery

Announcing services via Zeroconf/mDNS is an essential feature of openmediavault. A service will be automatically announced if it is enabled. This applies to NFS, FTP, Rsync, SMB/CIFS, SSH or the Workbench UI for example. Plugins may also announce their service via Zeroconf/mDNS.

As this is an essential feature, this cannot be switched off in general. However, it is possible to deactivate the advertising of individual services using *environment variables*.

To get a list of supported environment variables, run the following command:

```
# omv-env list | grep ZEROCONF_ENABLED
```

To enable or disable a service use this command:

```
# omv-env set -- OMV_XXX_ZEROCONF_ENABLED [yes|true|1|no]
```

Finally the modified *environment variable(s)* must be applied by running:

```
# omv-salt stage run prepare
# omv-salt deploy run avahi
```

Example:

```
# omv-env set -- OMV_PROFTPD_ZEROCONF_ENABLED no
# omv-salt stage run prepare
# omv-salt deploy run avahi
```

#### Firewall

This data table is for adding iptables rules. This can be useful if you need to secure access in your local network. Currently it is only possible to add rules to the OUTPUT and INPUT chains in the filter table. The configuration to load the rules at boot or network restart is done by the systemd unit called *openmediavault-firewall*.

**Tip**

- To avoid locking yourself out while testing, create a cron command to run every five minutes that flushes the OUTPUT/INPUT chain. Don't forget to delete the cron job after testing.:

```
*/5 * * * * root /sbin/iptables -F INPUT && /sbin/iptables -F OUTPUT
```

- Before adding the last rule to reject all, add a rule before the reject all, to LOG everything. This will help understand why some rules do not work. The log is saved in dmesg or syslog.

**Tip**

When seeking support please avoid posting screenshots of the data table, this is useless because it does not give the full overview of your firewall ruleset. Instead use:

```
$ iptables-save > /tmp/file.txt
```

### 5.1.3 Notifications

Notifications work in the form of email. The backend software used here is postfix[1] configured as a MTA in satellite mode. The options allow to configure to send mail via SMTP servers using the standard port or use SSL/TLS. The web interface allows the configuration of two recipient addresses. Both are assigned to the root user.

#### Configuration

The central MTA configuration is stored in `/etc/postfix/main.cf`

openmediavault creates the `/etc/postfix/recipient_canonical` to define the root (admin) and normal users mail addresses when added via the web interface. Example:

```
root rootthe@gmail.com
mike mikeadmin@themailco.com
@server.lan rootthe@gmail.com
```

When a scheduled task is defined to run as a certain user the output generated from that task, will be sent to that user defined mail.

The last line is the catch all address. For example a scheduled task set to be run as user with no mail defined in their profile will get the output generated sent to the catch all address (`rootthe@gmail.com`). The same will happen with any other mail action intended for an undefined user (not in that list).

Mails can be sent from terminal also with mail command. **`mail`** receives from stdin.

Examples 1:

```
$ echo "Message body" | mail -s "Test subject" mike
```

Mail will be delivered to `mikeadmin@themailco.com` as it is defined in canonical_recipients. The delivery address can be explicit also:

```
$ echo "Message body" | mail -s "Test subject" mikeadmin@themailco.com
```

Examples 2:

```
$ echo "Message body" | mail -s "Test subject" john
```

Mail will delivered to `rootthe@gmail.com` because user **john** does not have an email address defined in canonical_recipients, so it goes to the catch all address.

[1] http://www.postfix.org

**Warning**

openmediavault stores the configuration values in the database (including the password). Before posting information for support please sanitize the values.

### Events

The server will send notifications for this events:

- Log in from browser (If cookies are allowed, then it just sends once).
- Use of sudo by a user not in allowed group.
- Summary of locked users by pam_tally2[2]. This happens when a user or admin attempts fails to log in for more than three times.
- MD RAID events: degraded, reshape, etc. [D]
- Monit software: php-fpm, nginx, netatalk, rrdcached, collectd and omv-engined. [D]
- Monit filesystem: usage and mount points. [D]
- Monit system: CPU, Load and memory usage. [D]
- Scheduled tasks. [D]
- Rsync jobs. [D]
- Cron-apt: Summary of upgrade packages available. [D]
- SMART: Report of attribute changes. [D]

Options marked with [D] can be disabled selectively. The rest only when the whole notification backend is disabled.

### Gmail

Gmail can be used to send notifications, but only if you have enabled 2FA for your account. The App-Password must then be used as the password in the settings page. You can manage your Google App-Passwords here.

Please use the following settings:

```
SMTP Server: smtp.gmail.com
SMTP Port: 587
Encryption mode: STARTTLS
Sender email: <USERNAME>@gmail.com
Authentication required: Yes
Username: <USERNAME>@gmail.com
Password: <THE_APP_PASSWORD>
Primary email: <USERNAME>@gmail.com
Secondary email: optional
```

You can use any email address as your primary address. It can, but does not have to be, the sender address.

**Note**

Aliases are allowed. This is good for filtering later in gmail. <USERNAME>@gmail.com can be <USERNAME>+server1@gmail.com or <USERNAME>+whatever@gmail.com.

[2] http://www.linux-pam.org/Linux-PAM-html/sag-pam_tally2.html

**Note**

Google removed the "less secure apps" option on May 30, 2022. This means that there is no way to turn it on. An App-Password must be used instead as described above.

#### Third Party Notifications

Whenever a mail is dispatched by the MTA, postfix will execute a run-parts of this directory `/usr/share/openmediavault/notification/sink.d`, passing the following environmental variables:

```
OMV_NOTIFICATION_FROM
OMV_NOTIFICATION_RECIPIENT
OMV_NOTIFICATION_SUBJECT
OMV_NOTIFICATION_DATE
OMV_NOTIFICATION_MESSAGE_FILE
```

Also the following positional arguments are passed:

```
$1 The path of the file containing the message text (OMV_NOTIFICATION_MESSAGE_FILE)
$2 The FROM email address (OMV_NOTIFICATION_FROM)
$3 The TO recipient email adresses (OMV_NOTIFICATION_RECIPIENT)
```

Most modern non mail notifications systems have a documented API, where you can send text using curl payloads with a secret *TOKEN*. So most common case would be to use *OMV_NOTIFICATION_MESSAGE_FILE* variable only in your script.

Your script's filename must adhere to the following standards:

- Must belong to one or more of the following namespaces:
  - The LANANA-assigned namespace (`^[a-z0-9]+$`)
  - The LSB hierarchical and reserved namespaces (`^_?([a-z0-9_.]+-)+[a-z0-9]+$`)
  - The Debian cron script namespace (`^[a-zA-Z0-9_-]+$`)
- Start with a number like this: `<##>pushnotification`

**Note**

- Do not add an extension to your script in the run-parts directory, otherwise it will get excluded.
- Make sure the script file is executable. In this case also make sure the script is not a symlink to a mounted filesystem with *noexec* flag.

### 5.1.4 Scheduled Jobs

#### Overview

You can configure common and repetitive command(s) or scripts in this section. Is based on cron using the `minute hour day Week month` crontab syntax[1]. Due to web framework limitation, ranges are not supported. If you need a range you can configure a task for each day or simply use terminal with:

```
$ crontab -e
```

[1] https://linux.die.net/man/5/crontab

The grid panel reflects all current created cron jobs done via the web interface. The second field reflects the schedule in crontab language.

#### Options

**Username:** Under what user should the command/script be executed. You can select root, system accounts and openmediavault users.

**Mail Notification:** Send all the command/script output to the mail defined in the username profile. If the task is running as root, the mail recipient will be the one defined in notifications for primary and secondary delivery. If openmediavault user is defined in the task and has an email configured in their *profile* the notification will be sent to that mail address.

#### Configuration

The server configures all tasks done in the web interface creating this file `/etc/cron.d/openmediavault-userdefined` on demand as single lines per job.

```
SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin
# m h dom mon dow user    command
12 18 * * * root /var/lib/openmediavault/cron.d/userdefined-04dc9701-881f-4440-93e2-
↪66c385df4068 | mail -E -s "Cron - Movies" -a "From: Cron Daemon <root>" root >/dev/
↪null 2>&1
50 18 * * * root /var/lib/openmediavault/cron.d/userdefined-69a1cf21-3099-4d37-bb8f-
↪df3fecfac988 >/dev/null 2>&1
@daily root /var/lib/openmediavault/cron.d/userdefined-f04f0bbb-03d3-4d45-9efb-
↪e1e980cbbaf3 >/dev/null 2>&1
```

First is the cron time or interval, then username finally the command. The actual command is wrapped in a shell script located in this folder `/var/lib/openmediavault/cron.d/`. All files in there are prefixed with `username` and the internal database uuid.

**Warning**

- When using a single command to be executed, make sure this does not have any bashism. This because the cron wrapper script gets executed in pure shell #!/bin/sh. If you need to use something in bash wrap your command(s) in a bash script.
- @hourly, @daily, @weekly and @monthly are just nicknames. If you select @daily and your computer is shutdown at midnight the task will not run[Page 46, 1].

### 5.1.5 Power Management

#### Monitoring

Configures *cpufrequtils* and sets the default options for the governor to be **conservative** by default in x86 architectures if enabled. If architecture is different then governor is set as **ondemand**.

`/etc/default/cpufrequtils`

```
ENABLE="true"
GOVERNOR="conservative"
MAX_SPEED="0"
MIN_SPEED="0"
```

`/etc/default/loadcpufreq`

```
ENABLE="true"
```

All values above can be changed via *environmental variables*.

#### Power button

Configures the action to take when pressing the mechanical power button of the server.

#### Scheduled

Based on cron, is possible to define shutdown, hibernation or suspend times for the server.

### 5.1.6 Certificates

This section allows you to create or import SSH keys or SSL certificates.

#### SSH (Secure Shell)

The public/private pair keys created or imported here are for using in the *RSync client (jobs)* service section. Plugins can use the internal database if they want to use these keys using the SSH certificates combo class. The key pair will be stored in the internal database, but only the public key will be available for display just by clicking edit. Not displaying the private key is basic ssh security as it never has to leave the host where it was created. The public key can be copied to clipboard or any other transport to be added to a remote server. Add a comment as this will be appended to the public key, this is important if you need to revoke the key pair in the remote server in case the server that generated the pair is compromised. The keys are stored beside the database in these two files:

- **Public key**: `/etc/ssh/openmediavault-<uuid_suffix>.pub`
- **Private key**: `/etc/ssh/openmediavault-<uuid_suffix>`

The <uuid> suffix is the internal openmediavault reference number.

> **Note**
>
> The public key is not displayed in RFC 4716. In case the remote server is also openmediavault based, you need to *convert* it the appropiate format.

#### SSL (Secure Socket Layer)

The SSL certificates created or imported here can be used by the web interface or FTP server. Plugins can also use them by adding the SSL certificate combo class. The create window has the most common SSL certificates fields. The certificate/private pair is stored in the internal database and as files in the Linux standard SSL location. Certificate file with a <uuid> suffix, which is the internal database number:

`/etc/ssl/certs/openmediavault-<uuid>.cert`

Private key file with the same <uuid> suffix as their certificate pair.

`/etc/ssl/private/openmediavault-<uuid>.key`

When importing existing ssl certificates make sure they are formatted/converted appropriately.

The command that creates the certificate runs in the PHP backend and is documented here. This certificates are self signed, without root CA.

#### LetsEncrypt

Lets Encrypt certificates can be imported directly, just locate your `/etc/letsencrypt/live/<mydomain.com>/fullchain,privkey.pem` files and copy their contents in their respective field. No need to convert.

## 5.2 Storage

### 5.2.1 Disks

An overview of all physical disks attached to the server. Displays basic information to identify disks, such as: manufacturer, model, serial number and capacity. A hidden column also displays the linux block device identification symlinks `/dev/disk/by-id,by-path,by-uuid`.

Be aware that when attaching disks via USB (a docking station, cage, adapter, etc.) the internal disk information will not pass, the backend will display probably the USB-SATA controller information. The capacity should remain the same. This is a response given by the backend with `DiskMgnt::getList` service-method using a rock64 SBC board with a docking station attached via the USB 3.0 port:


```
{
   "response":{
      "total":3,
      "data":[
         {
            "devicename":"mmcblk1",
            "devicefile":"/dev/mmcblk1",
            "devicelinks":[
               "/dev/disk/by-id/mmc-SL16G_0x0091d901",
               "/dev/disk/by-path/platform-ff500000.dwmmc"
            ],
            "model":"",
            "size":"15931539456",
            "description":"n/a [/dev/mmcblk1, 14.83 GiB]",
            "vendor":"",
            "serialnumber":"",
            "israid":false,
            "isroot":true
         },
         {
            "devicename":"sda",
            "devicefile":"/dev/sda",
            "devicelinks":[
               "/dev/disk/by-path/platform-xhci-hcd.8.auto-usb-0:1:1.0-scsi-0:0:0:0",
               "/dev/disk/by-id/usb-USB_3.0_HDD_Docking_Station_2017101701E0-0:0"
            ],
            "model":"",
            "size":"500107862016",
            "description":"n/a [/dev/sda, 465.76 GiB]",
            "vendor":"USB 3.0",
            "serialnumber":"2017101701E0",
            "israid":false,
            "isroot":false
         },
         {
            "devicename":"sdb",
```


(continues on next page)

(continued from previous page)


```
            "devicefile":"/dev/sdb",
            "devicelinks":[
               "/dev/disk/by-id/usb-USB_3.0_HDD_Docking_Station_2017101701E0-0:1",
               "/dev/disk/by-path/platform-xhci-hcd.8.auto-usb-0:1:1.0-scsi-0:0:0:1"
            ],
            "model":"",
            "size":"2000398934016",
            "description":"n/a [/dev/sdb, 1.81 TiB]",
            "vendor":"USB 3.0",
            "serialnumber":"2017101701E0",
            "israid":false,
            "isroot":false
         }
      ]
   },
   "error":null
}
```


Notice here `sdb` and `sda` both disks show same serial number and that is incorrect. There is no vendor and model shows as "USB 3.0".

In this cases you can access the disk information in the SMART section, not the grid but the information button. External portable USB hard drives should display information normally.

### Power Options

Pressing the edit button with a selected disk will give the following options available to set:

- Advanced power management (APM)
- Automatic accoustic management (Not all drives support this)
- Spindown time (ST)
- Write cache

All the above options are configured using smartmontools[1]. The APM values from the interface are resumed in seven steps with a small description to make it easier for the user to select. If you want to experiment with intermediate values that are not supported via the UI, then you can edit the `/etc/openmediavault/config.xml` by editing the XPath `/system/storage/hdparm`, change the values for the disk, and finally run:

```
$ omv-salt deploy run smartmontools
```

When setting a spin down time make sure APM is set bellow 128, otherwise it will not work. The UI does not narrow the APM options if spin down time is set, or disables the spin down menu when a value higher than 128 is selected.

**Note**

For changes to be permanent, settings are stored in files that are located in `/etc/smartmontools/hdparm.d/`. You can add custom files there according to the README. To apply the settings, run the command `systemctl restart smartctl-hdparm.service`.

[1] https://www.smartmontools.org/

#### Wipe

If you need to erase data from your disks, you can use this button. It gives the secure or quick option.

The quick option basically erases the partition table and signatures (MBR or GPT) by using this command:

```
$ sgdisk --zap-all /dev/sdX
```

After that it ensures is clean by using dd:

```
$ dd if=/dev/zero of=/dev/sdX bs=4M count=1
```

Which erases the beginning of the disk.

The secure mode will rewrite the block device entirely. This process takes a long time and is only one iteration. It uses this command:

```
$ shred -v -n 1 /dev/sdX
```

### 5.2.2 SMART

Modern hard disks drives (and SSD's) have firmware inside that reports several attributes (usually called S.M.A.R.T values) through sensors that are relevant to assess the device condition. Those values and what they mean are explained by here. Not all drives report the same amount of attributes, but all of them report some common ones which are known to be the best for assessing health[1].

There are several tools for accessing those attributes. openmediavault reads and monitors hard drives smart values using smartmontools[2].

Notifications are integrated with smartmontools. Changes in S.M.A.R.T values are reported via mail.

> **Note**
>
> Please note that SMART is not supported by openmediavault for disks that are assembled as a RAID via HBA. openmediavault only supports SMART for those disks that are connected to an HBA in pass-through mode.

#### General

This enables smartd (SMART daemon). The daemon will periodically check disks attributes and compare them with previous check. You can select the daemon not to poll information if the disks are in certain power state.

Temperature is a very critical attribute. Select the desired limits for smart monitoring to report on changes[3].

#### Devices

The grid displays all current block devices in the system with SMART capabilities. From this grid by selecting a drive you can configure if you want smartmontools to watch and inform for any SMART attributes changes during uptime using the edit button.

The information button displays several tabs which provide friendly parsed information about the drive. The last tab has all the information in raw text.

The grid columns shows different identification values for the drive, the last one (Status) reports a green icon if drive is in good condition or red if drive needs some attention, if you hover on the icon a tooltip that will report more details.

[1] https://www.backblaze.com/blog/what-smart-stats-indicate-hard-drive-failures/
[2] https://www.smartmontools.org/
[3] https://www.backblaze.com/blog/hard-drive-temperature-does-it-matter/

The code that reports the red icon is based on this function here and here, so basically the red icon will be triggered only on attributes with the prefailure (P) flag when:

- Any attribute (P) current value is equal or less than threshold –> Bad attribute now
- The attribute (P) worst value is equal or less the threshold value –> Bad attribute in the past
- Reallocated_Sector_Ct (`id=5`) and Current_Pending_Sector (`id=197) raw attributes values report more than 1 –> Bad sector

**Note**

Do not enable SMART for a virtual block device (Virtualbox, Proxmox or ESXi) and expect to get reports of SMART values.

#### Scheduled tests

Gives an option to select four different scheduled tests:

- Short self-test
- Long self-test
- Conveyance self-test
- Offline immediate test

These tests and what they do are explained here and here.

SMART only reallocates a bad sector on write. A manual method to force reallocating the pending(s) sector(s) is described here.

### 5.2.3 RAID

openmediavault uses linux software RAID driver (MD) and the mdadm utility to create arrays[1]. Arrays created in any other linux distro should be recognized immediately by the server. In most cases you can skip to the filesystem array and proceed to mount to integrate the filesystem into the database.

#### Overview

The grid panel shows all currently available MD arrays. There is no internal database section for RAID arrays, so every array that is assembled in the server should be displayed here.

**Create**

The following table displays levels available in the web interface:

Table 1: RAID

| Level | Name | Min. Disks | Redundancy | Growable |
|---|---|---|---|---|
| Linear | JBOD | 2 | No | No |
| 0 | Stripe | 2 | No | No |
| 1 | Mirror | 2 | Yes | Yes |
| 5 | RAID5 | 3 | Yes | Yes |
| 6 | RAID6 | 4 | Yes | Yes |
| 10 | Stripped Mirror | 4 | Yes | No |

[1] https://raid.wiki.kernel.org/index.php/Linux_Raid

**Note**

- RAID4 and FAULTY levels are not supported in the web interface.
- RAID1+0 is possible by stripping two mirrors. The create window should display both mirrors if they do not have any filesystem signatures yet.

**Detail**
Displays extended information of the array, the output comes from **mdadm -detail /dev/mdX**

**Grow**
Add disk(s) into the array.

**Recover**
If the array comes from another linux server you can use this button to reassemble the array in the current server

**Remove**
This is used to remove failed disks, in case one needs be replaced.

**Delete**
Stop the array and zero the superblock of all devices conforming the array (script **/usr/sbin/omv-rmraid**). Use with caution.

Mdadm works better with unpartitioned disks, plain raw block devices. Before creating MD RAID in your system make sure disks are clean before. In the physical disk section you can perform a quick or full wipe. Quick wipe is enough to delete partition tables.

Degraded array creation in not possible in the web interface, however the array can be created in terminal using mdadm if you want for example to convert a RAID from level 1 to 5 or 6.

Mail notifications are integrated for mdadm, these are sent everytime an array enters degraded state.

### Growing

Before growing array is better to clean the partition table of the new disk, specially if the disk was used before in another mdadm array, erase also the superblock:

```
$ mdadm --zero-superblock /dev/sdX
```

After adding a disk to the array, the re-synching process will begin immediately. Depending on the size of the disks this process can take several hours or even days, this is because mdadm tries to balance resources and keep usability of the system not using high CPU and RAM. To speed the process:

```
$ echo ${value} > /proc/sys/dev/raid/speed_limit_min #value is interpreted as kbytes/
↪seconds
```

After synching is finish is necessary to expand the filesystem using the resize button.

**Warning**

- If your MD array and filesystem was created with openmediavault or Debian before December 2014, then please read *here*.
- Do not use RAID arrays in production with drives connected via USB, neither hubs or different ports. This includes low power devices that do not have a SATA controller, e.g. Raspberry Pi, Pogoplugs and any low entry ARM SBC.

### 5.2.4 Filesystems

**Overview**

The filesystem section of the openmediavault web interface is where you integrate disk volumes to be part of the server. Drives/filesystems that are not mounted through the web interface are not registered in the backend database, this means you cannot use volumes to create shared folders if they were not mounted properly. *This is very important*, users that come from an existing debian installation with filesystems already present in their fstab file will see that no volumes will be available for creating shared folders even if they are mounted. For the disks to be properly integrated it is better to delete all fstab lines except rootfs and swap, reboot your server and start mounting the disks through the web interface.

The mount process acts like many other services in openmediavault, first it writes a database entry in `config.xml`, this entry contains essential information:

- UUID of the database object <*uuid*>
- Predictable device path of the filesystem <*fsname*>
- Target mount directory <*dir*>
- Filesystem options <*opts*>
- Filesystem type (EXT3, EXT4, etc.) <*type*>

You can inspect a *mntent* entry in `config.xml` located in `/etc/openmediavault/`, it should look like this:

```
<mntent>
        <uuid>f767ee54-eb3a-44c5-b159-1840a289c84b</uuid>
        <fsname>/dev/disk/by-label/VOLUME1</fsname>
        <dir>/srv/dev-disk-by-label-VOLUME1</dir>
        <type>ext4</type>
        <opts>defaults,nofail,user_xattr,usrjquota=aquota.user,grpjquota=aquota.
↪group,jqfmt=vfsv0,acl</opts>
        <freq>0</freq>
        <passno>2</passno>
        <hidden>0</hidden>
</mntent>
```

With the *mntent* entry in `config.xml`, **`omv-salt deploy run fstab`** script writes the appropriate line in `/etc/fstab`. You can identify entries in `/etc/fstab` created by the web interface by looking at «openmediavault» tags. It is important to mention to not alter the information in between these tags. If you delete or modify a fstab option (*noexec* or *quota* for example) the next time you mount a new disk into the server, **`omv-salt deploy run fstab`** will deploy the original value there again. If you need persistent change use environmental variables. Finally the backend will proceed to mount the filesystem. After this the volume is ready for creating shared folders.

**Resize**

The resize button is used for expanding filesystems. This can occur if you decide to resize a disk partition or you have grown a RAID array by adding one or more disks.

**Delete**

The delete button actually deletes filesystems, using **`wipefs -a`**. This will flush filesystem, raid or partition-table signatures (magic strings). Be careful using this. The button is disabled until the filesystem is actually unmounted.

**Unmount**

Disabled until you have deleted all shared folders associated with that volume. Unmount will remove the entry from `config.xml` and `/etc/fstab`.

**Supported Filesystems**

openmediavault supports the following filesystems that can be mounted through the web interface:

Table 2: openmediavault supported filesystems

| Type | Format | Mount |
|---|---|---|
| btrfs | yes | yes |
| exfat | no | yes |
| ext3 | yes | yes |
| ext4 | yes | yes |
| f2fs | yes | yes |
| hfsplus | no | yes |
| iso9660 | no | yes |
| jfs | yes | yes |
| msdos | no | yes |
| ntfs | no | yes |
| udf | no | yes |
| ufs | no | yes |
| umsdos | no | yes |
| vfat | no | yes |
| xfs | yes | yes |
| zfs | no | no |

**Note**

**BTRFS**

- Creating multi device filesystems is not supported in the web interface. However you can add devices to your btrfs array in CLI and it will not present any problems.
- No extra features of btrfs are available in the webui like snapshots or subvolumes. Additional subvolumes will have either be mounted outside of the OMV fstab tags or manually add mntent entries to config.xml or use advanced configuration

**Note**

**ZFS**

Support for ZFS is available through ZoL and requires the use of a third party plugin provided by omv-extras. This includes code added to the OMV filesystem backend. The plugin allows you to create shared folders for ZFS volumes. On AMD64, if ZFS support is desired, it is recommended to first use omv-extras install the third-party openmediavault-kernel plugin and use that to install the ProxMox (PVE) kernel, which has precompiled support for ZFS and may improve stability. After that, install the ZFS plugin.

### 5.2.5 Shared Folders

The shared folder are the key functionality in openmediavault around which all services revolve. They will be created as subvolumes on BTRFS file systems or simple directories on all other file systems supported by openmediavault. If a shared folder is located on a BTRFS file system, then it is possible to create snapshots of it. This can be done manually or via scheduled tasks.

### Add

When a shared folder is created using the add button, the window form displays the following options:

- **Name:** The logical name. This can override the path name. Typing a name here will fill the path with the same string.
- **Device:** The parent filesystem associated with the shared folder.
- **Path:** The relative path to the mounted device. To share the whole disk just type `/`.
- **Permissions:** The default descriptive text will create the shared folder with `root:users` ownership and `775` permission mode.

**Available modes**

| Logical name | Octal mode |
|---|---|
| Administrator: read/write, Users: no access, Others: no access | 700 |
| Administrator: read/write, Users: read only, Others: no access | 750 |
| Administrator: read/write, Users: read/write, Everyone: no access | 770 |
| Administrator: read/write, Users: read only, Everyone: read-only | 755 |
| Administrator: read/write, Users: read/write, Everyone: read-only | 775 (Default) |
| Everyone: read/write | 777 |

This is how a shared folder looks inside the `config.xml` database:

```
<sharedfolder>
    <uuid>9535a292-11e2-4528-8ae2-e1be17cf1fde</uuid>
    <name>videos</name>
    <comment></comment>
    <mntentref>4adf0892-cf63-466f-a5aa-80a152b8dea6</mntentref>
    <reldirpath>data/videos/</reldirpath>
    <privileges>
      <privilege>
        <type>user</type>
        <name>john</name>
        <perms>7</perms>
      </privilege>
      <privilege>
        <type>user</type>
        <name>mike</name>
        <perms>5</perms>
      </privilege>
    </privileges>
</sharedfolder>
```

Some of the elements explained:

- **uuid**: Internal database reference number.
- **name**: logical name given to the shared folder.
- **mntentref**: the associated filesystem reference. The number is in the `uuid` format, the fstab section in `config.xml` should contain a `<mntent>` reference with this number.
- **reldirpath**: Path relative to the parent filesystem.
- **privileges**: Users associated with the shared folder and their access level.

When a plugin or a service uses a shared folder it stores the uuid value only. Later on using helper scripts or internal openmediavault functions the full path can be obtained just by using the `uuid`. An example in shell:

```
$ . /usr/share/openmediavault/scripts/helper-functions && omv_get_sharedfolder_path␣
↪9535a292-11e2-4528-8ae2-e1be17cf1fde
```

This returns:

```
$ /srv/dev-disk-by-label-VOLUME1/data_general/videos
```

More information about helper functions.

A shared folder can be used across all over the system backend. Is available to select it in sharing services (FTP, Samba, RSync, etc.) at the same time. Plugins can use them also just by using the shared folder combo class.

**Note**

- A shared folder belongs to an internal openmediavault database filesystem entry. Is not possible to unmount the filesystem without deleting the folder configuration from the web interface.
- If a shared folder is being used by a service (FTP, plugins, etc.) is not possible to delete it. Is necessary to disengage the shared folder from the service(s) or section(s) that is holding it before proceeding with removal. This will also prevent to unmount a device from the web interface in the filesystem section if there is still a shared folder associated with it.
- Due to the design of the software is not possible at the moment to know what section or service is holding which shared folder.

### Edit

Edit shared folder is possible, but it has some limitations. You can only change the parent device volume. Once the parent device is changed the backend will reconfigure every service that is using a shared folder and stop/start daemons accordingly.

Be aware that changing the parent device volume will not move the data from one filesystem to another.

**Warning**

**NFS Server**: Editing the parent device will not descent into `/etc/fstab`. Make sure you edit the share in the NFS section so the bind can be remounted.

### Permissions

Set the shared folder's read and write permissions for users and groups. These settings are used by the different services (SMB, FTP and AFP). They have no effect on the permissions of the file system. It will display all the openmediavault users/groups and their corresponding permissions for the selected shared folder.

As you can see in the *database example*, permissions are expressed in the internal database in the same manner as permissions in Linux, simplified using the octal mode: *read/write(7)*, *read-only(5) and no access(0)*.

Permissions can be edited per *shared folder* or *user*.

If a permission is changed, it means a change in the shared folder database section. This database event will trigger a reconfiguration of SMB, FTP and AFP, and it will also restart all the preceding daemons. A shared folder service not using the permission information from the database entry does not get reconfigured/restarted if only a permission change occurs.

#### Access Control List (ACL)

Provides fine grained permission control besides the standard POSIX permissions. The usage of ACL is not recommended for the average home user. If a server is using an extensive list of users then ACL could suit better[1][2].

The expanded ACL window displays three panels. Left one is a browser of the selected shared folder, so you can see the apply ACL to the current folder or a subdirectory and so on.

The left panel displays all current openmediavault users and system accounts and their current ACL of the selected folder. This panel actually reads ACL from the selected folder.

The bottom panel displays the standard POSIX permission of the selected folder or subfolders in a user friendly interface.

If you want just to reset linux permissions, just use the recursive checkbox and change options only in the bottom panel, and not selecting any ACL user/group in left panel.

The ACL is applied using **`setfacl`**[3] and read with **`getfacl`**[4].

**Note**

- openmediavault mounts all Linux filesystems with ACL enabled. Only native linux POSIX filesystems support ACL. The button gets disabled for HFS+, NTFS, FAT, etc.
- ZFS provides ACL support, just need to enable the pool/dataset property.

## 5.3 Services

### 5.3.1 Samba

Samba server comes from Debian software repositories. The openmediavault project does not maintain this package, all bugs, hotfixes and features come from Debian.

#### General

The server configures Samba as standalone mode. The default global section

```
[global]
workgroup = HOME
server string = %h server
dns proxy = no
log level = 0
syslog = 0
log file = /var/log/samba/log.%m
max log size = 1000
syslog only = yes
panic action = /usr/share/samba/panic-action %d
encrypt passwords = true
passdb backend = tdbsam
obey pam restrictions = no
unix password sync = no
passwd program = /usr/bin/passwd %u
```

(continues on next page)

[1] https://help.ubuntu.com/community/FilePermissionsACLs
[2] http://vanemery.net/Linux/ACL/linux-acl.html
[3] https://linux.die.net/man/1/setfacl
[4] https://linux.die.net/man/1/getfacl

(continued from previous page)

```
passwd chat = *Enter\snew\s*\spassword:* %n\n *Retype\snew\s*\spassword:* %n\n *password\
↪supdated\ssuccessfully* .
pam password change = yes
socket options = TCP_NODELAY IPTOS_LOWDELAY
guest account = nobody
load printers = no
disable spoolss = yes
printing = bsd
printcap name = /dev/null
unix extensions = yes
wide links = no
create mask = 0777
directory mask = 0777
use sendfile = yes
aio read size = 16384
aio write size = 16384
null passwords = no
local master = yes
time server = no
wins support = no
```

A default share example:

```
[MyDocuments]
path = /media//dev/disk/by-label/VOLUME1/Documents/
guest ok = no
read only = no
browseable = yes
inherit acls = yes
inherit permissions = no
ea support = no
store dos attributes = no
printable = no
create mask = 0755
force create mode = 0644
directory mask = 0755
force directory mode = 0755
hide dot files = yes
valid users = "john"
invalid users =
read list =
write list = "john"
```

openmediavault automatically configures shadow copies if the shared folder is on a BTRFS file system and if there are snapshots available. These are exposed as *Previous Versions* to Windows clients.

It is possible to add extra options in the general and share configuration page. Options that are managed by openmediavault can be customized via *environmental variables*.

### Permissions

The login access in Samba is configured using permissions. This means they will not act in the file system layer, they will run in the Samba authentication layer. From there the access can be controlled to be read only or read/write access and guest account access. This is done with the `Permissions` button in the shared folder section, not the ACL. Permissions only gets login access and from there determines if user can read or write. If write access is enabled but files/folders have restricted permissions then write access is not possible using Samba.

**Important**

Samba does not use PAM for login, it has a different password database. When the admin changes a username password (or the user changes their own) using the web interface what openmediavault does is that it changes both the linux login password and the Samba internal database. If a username changes their password using shell, this will not be reflected in Samba log in.

### Share types

**Non-public (Private):** *Login always required, Guest Allowed denied*

```
guest ok = no
valid users = User1, User2, @Group1, @Group2 ## this will deny all none authorized users
read list = User1, @Group1
write list = User2, @Group2
```

This means that every user will have to provide valid OMV credentials to access that share. Also this type of shares requires at least one definition of a valid user, otherwise the directive would be empty.

**Note**

This will allow every user to log into the share.

**Semi-public:** *When login is not provided, the guest user is used. This is the "guest allowed" option from the Samba share option*

```
guest ok = yes
read list = User1, @Group1
write list = User2, @Group2
```

Notice here if users are not set up permissions (that means blank tick boxes) anyone will be able to login anyway and have write access.

**Public only:** *The guest user is always used. This is the Guest Only option in the Samba share configuration.*

```
guest ok = yes
guest only = yes
```

With these options valid, read only and write user directives will be ignored when the `/etc/samba/smb.conf` is deployed by openmediavault.

**Note**

- The guest account is mapped to system account nobody, it doesn't belong to group users, thus it has, by default, NO WRITE ACCESS just READ. This can be reverted modifying the POSIX permissions of the share to 777.
- These directives are NOT ACL.

#### Questions

**How do I enter credentials in a semi-public share?**
In most cases the user will always be logged as guest. Use Windows map network drive feature to provide other login credentials different from guest. In Mac OS X use CMD+K (if using Finder)

**Why the login keeps saying access denied?**

**This is more likely caused by two things:**

- Permission issue (ACL or non default POSIX permission mode/ownership). Fix the permissions in the shared folder. Samba runs as privileged (root) user, even so if parts of path don't have adecuate permissions, it will still respond access denied.
- Out of sync password in between linux and Samba. This is very rare but it has happened. Test in ssh the following [tt]smbpasswd username[/tt] enter password and try and login again.

**Why I can't edit files that other users have created?**
The default umask in Samba is `644` for files. To enable flexible sharing check *Enable permission inheritance* in the Samba share settings, this will force `664` creation mode. Files created previously need to change their permission mode. Check also that you don't have read only enabled. This option overrides privileges and POSIX.

### 5.3.2 FTP

#### Overview

On top of the proftpd debian package, openmediavault uses the vroot module by Castaglia. The server is configured using a DefaultRoot for this folder `/srv/ftp`. Adding folders to the chroot is done by using vroot aliases. This is the default behaviour of the FTP server and cannot be changed. The vroot default path can be changed with environmental variables. The chroot also prevents symlinks from escaping that path, however you can use symlinks that point inside the chroot. So any time you add a shared folder to the FTP, OMV will create first a vroot alias:

```
<IfModule mod_vroot.c>
  VRootAlias "/media/dev-disk-by-label-VOLUME1/videos" "Videos"
</IfModule>
```

Then that alias will have privileges assigned:

```
<Directory /Videos>
  <Limit ALL>
    AllowUser OR omvUser
    DenyAll
  </Limit>
  <Limit READ DIRS>
    AllowUser OR omvUser
    DenyAll
  </Limit>
</Directory>
```

By default you're not allowed to write in there when you login, this means you cannot create folders in the landing directory, you have to enter one of the shared folders. Also due to the nature of the chroot, creating top level folders is

pointless since they will be actually stored in `/srv/ftp` and not in the media disks.

### Remote Access

FTP is a protocol intended for use in LAN and WAN. For accessing WAN it is required to forward the server port (default 21) and the passive range to the openmediavault server.

### Anonymous Login

Disabled by default, the anonymous user is mapped to the system user ftp and nogroup. There is no write access for anonymous and this is configured in the `/etc/proftpd/proftpd.conf` file and cannot be changed as is hard coded into the default configuration script of the server. In this case there is no environmental variable to change that behaviour:

```
<Anonymous ~ftp>
  User ftp
  Group nogroup
  UserAlias anonymous ftp
  DirFakeUser on ftp
  DirFakeGroup on ftp
  RequireValidShell off
  <Directory *>
    HideFiles (welcome.msg)
    HideNoAccess on
    <Limit WRITE>
      DenyAll
    </Limit>
  </Directory>
</Anonymous>
```

### FTP(S/ES)

openmediavault provides two SSL/TLS modes for encrypting the FTP communication, implicit and explicit FTPS. The differences and features are explained here and here. Enabling FTP over SSL/TLS requires first that you create or import a certificate in the corresponding section. Once the certficate is there you can choose it from SSL/TLS section in FTP. The default FTPS of the server is explicit, you can click the checkbox to enable implicit. If you choose implicit make sure you forward port 900 in your router to port 21 in your NAS server if you're accessing from WAN, otherwise the client will probably display ECONREFUSED.

### Tips

**Login Group**

By default all openmediavault users created in the web interface can log into FTP. You can restrict to read only or read write, there is no deny access, but a user without privileges would not see that folder. If you want to add a layer of extra security for the login, you can create a control group to restrict login to FTP. You first create a group, for example ftp_users, then at the end of the general extra options of the server we add:

```
<Limit LOGIN>
    DenyGroup !ftp_users
</Limit>
```

Only users members of that particular group will be able to log into the FTP server.

**Home Folders**

There is no straightforward way of doing this in the web interface, but if you really need home folders for FTP, you can change the default vroot path with environmental variable `OMV_PROFTPD_MODAUTH_DEFAULTROOT="~"`.

What will happen here is users will log in straight into their home folders. If you add shared folders to the server they will be displayed inside the user home folder plus any other folder present in their home folder.

**LetsEncrypt**
Just import your LE certificate in the `General | Certificates | SSL` *section*. Then in the TLS/SSL tab, select the imported cert from the dropdown menu. Do not enable implicit ssl. You need also to add the chain file. So in the extra option field text add:

`TLSCACertificateFile <yourpathtoLE>/etc/letsencrypt/live/<yourdomain>/chain.pem`

### 5.3.3 NFS

#### Overview

The configuration of the server is done using the common NFS guidelines. Shared folders are actually binded to the /export directory. You can check by examining the `/etc/fstab` file after you have added a folder to the server. All NFS server configured folders are in /etc/exports as follows:

```
/export/Shared_1 (fsid=1,rw,subtree_check,secure,root_squash)
/export/Videos 10.10.0.0/24 (fsid=2,rw,subtree_check,secure,nroot_squash)
/export (ro,fsid=0,root_squash,no_subtree_check,hide)
```

The first two lines are examples, the last line is the NFSv4 pseudo file system.[1][2]

#### Server Shares

The following options are available to configure from the web interface:

- **Shared folder:** Select a folder, the system will add an bind entry to fstab, mount that bind and add it to /etc/exports file
- **Client:** Enter a single IP, host, network in CIDR notation or wildcards. Only one entry is allowed. Check the manual about the machine name formats.
- **Privilege:** This will append read write (rw) or read-only (ro) to `/etc/exports`.[3]
- **Extra options:** Add options according the manual.

The server also shares by default the pseudo root filesystem of /exports as NFSv4.

#### Clients

To access NFS shares using any debian derived linux distro:

- Mount as NFSv4 all folders in `/export/` in `/mnt/nfs`:

```
$ mount 172.34.3.12:/ /mnt/nfs
```

- Mount as NFSv3 all folders inside `/export` in `/mnt/nfs`:

```
$ mount 172.34.3.12:/export /mnt/nfs
```

- Mount as NFSv3 the folder `/export/Videos` in `/mnt/nfs`:

```
$ mount 172.34.3.12:/export/Videos /mnt/nfs
```

- Mount as NFSv4 the folder `/export/Videos` in `/mnt/nfs`:

[1] https://help.ubuntu.com/community/NFSv4Howto#NFSv4_without_Kerberos
[2] https://www.centos.org/docs/5/html/5.1/Deployment_Guide/s3-nfs-server-config-exportfs-nfsv4.html
[3] This is not standard openmediavault privileges as in the shared folder section.

```
$ mount 172.34.3.12:/Videos /mnt/nfs
```

Check your distro on how to proceed with different NFS versions.

#### NFSv4 Pseudo filesystem

The default /export folder is shared with this default options `ro,wdelay,root_squash,no_subtree_check,fsid=0` only available to change via environmental variables, so be aware that mounting this path you will encounter permission problems.

#### Permissions

NFS relies on uid/gid matching at the remote/local filesystem and it doesn't provide any authentication/security at all. Basic security is provided by using network allow, and squash options. If you want extra security in NFS, you will need to configure it to use kerberos ticketing system.

#### Tips

**Macos/OSX**
: If you want to mount your NFS exports, add insecure in extra options or use `resvport` in the command line.

  Example:

```
$ sudo mount -t nfs -o resvport,rw 192.168.3.1:/export/Videos /private/nfs
```

**Debian**
: Debian distributions (and many others) always include the group users with `gid=100` by default, if you want to resolve permissions easily for all users of a PC using linux add `anongid=100` in extra options. This will force all mounts to use that gid.

**Symlinks**
: This are not followed outside of their export path, so they have to be relative.

**Remote access**
: NFS was designed to be used as a local network protocol. Do not expose the NFS server to the Internet. If you still need access use a VPN.

### 5.3.4 SSH

#### Overview

Secure shell comes enabled by default in openmediavault.

> **Note**
>
> openmediavault will enable SSH access for the user `root` by default to be able to access a headless system in case of a broken installation or other maintenance situations. You should disable this behaviour in the `Services | SSH` page for security reasons after installation.
>
> To still get `root` access you need to create a non-privileged user and add them to the `_ssh` and `sudo` groups. After that you can SSH into the system with this non-privileged user and run `sudo su`.

The configuration options via web interface are minimal:

- Disable the root login
- Disable password authentication

- Enable public key authentication (PKA)
- Enable compression
- Enable tunneling (for SOCKS and port forward)

An extra options field is provided to enter more options. Examine first the file `/etc/ssh/sshd_config` before adding extra options otherwise the option will not be applied. You may also check the SSH related *environmental variables* that can be used to customize several options.

Normal users created in the web interface can access the remote shell by adding them to the `_ssh` group. Using PKA for users requires keys to be added to their profile. This is described in the *Users* section. The public key has to be added in *OpenSSH* or RFC 4716 format.

To convert a public key run:

```
$ ssh-keygen -e -f nameofthekey.pub
```

Paste the output in the users profile at `Users | Users | <USERNAME> | Edit | Public Keys`.

A public key in RFC 4716 looks like this:

```
---- BEGIN SSH2 PUBLIC KEY ----
Comment: "iPhone user1"
AAAAB3NzaC1yc2EAAAADAQABAAABAQDfSQulxffUktx2P6EikkjVxDw0tT8nCW8LHLx/kl
8t37xFQ5/OoL9m6rVzYy5CFGYt+l7DffWjL0Av7AqaM0ykZVmv2VEM6TmMo56LTlmyZdlz
X5+GEJgCQNtDxcIYAVuPXKpLtlB/uAGzwHdZWpAen+mHgWIi4va8N5QNn4rXpkREcvM1q4
snyAi+gyjAS2Dn4pm8zzrd9qQFnoRYzidbp5e2Rs3brOkwUco0ZkOME2Ff6SpLGaXz4DHH
qgdTqZwHaTXEwm6kDmglCQrauIPI/ggNqz9mVEspYkskR2PM4CAty8RkZD4MQe5K3EMAFR
aFobBSlhQ3ESCYWNXTS3bF
---- END SSH2 PUBLIC KEY ----
```

The comment string is very important. This will help track down when is necessary to revoke the key in case it gets lost or stolen.

#### Admin Tasks

If root login has been disabled and need to perform administrative tasks in the terminal, swap to root by typing:

```
$ su
```

> **Note**
>
> **Remote WAN access**
>
> - Forward in router/firewall a port different than 22. This will minimize bots fingering the SSH server.
> - Always use PKA.
> - Disable password login.
> - Disable root login.

### 5.3.5 RSync

The server can be configured to act as a client to pull and push data to remote locations as well as act an RSync daemon server, where other clients can retrieve or store data from/to the server. In RSync language, the shared folders are called modules. Since openmediavault version 3.0 is possible now to create remote RSync jobs using ssh as transport shell. The RSync is divided in two tabs:

### Jobs (client)

Based on cron, the tasks can be configured to run at certain time or make it repetitive. A few of the options explained:

**Type**

- Local: This will run an RSync in between two internal folders of the server. For example you can use this to move data across different disks in your system
- Remote: This will deactivate destination folder, and instead you'll need to place a destination server address. You can select here:

  **Mode (remote)**

  - Push: store contents to a remote server
  - Pull: Retrieve contents from a remote server

Selecting one or the other will invert the folder as source or destination, the same as the server address.

**Destination/Source Shared Folder**
Choose a shared folder where you want the contents to be stored (pull) or you want the contents from that folder to be sent to a remote server (push).

**Destination/Source Server**
You need to put address server host or ip.

Examples:

If you are targeting the job against an RSync daemon server:

```
rsync://10.10.10.12/ModuleName
username@10.10.10.12::ModuleName
rsync://username@10.10.10.12:873/ModuleName
```

If you are going to connect to another server just using SSH with public key:

```
username@10.10.0.12:/srv/dev-disk-by-label-VOLUME1/Documents
```

**Warning**

When the RSync task is configured using ssh with PKA, the script that runs the jobs is non-interactive, this means there cannot be neither a passphrase for the private key nor a login password. Make sure your private key is not created with a password (in case is imported). Also make sure the remote server can accept PKA and not enforce password login.

**Authentication (remote)**

- **Password**: For the remote RSync daemon module. Is not the username login password defined in the Rights Management section of the server. Read ahead in server tab.
- **Public Key**: Select a key. These are created/imported from `General | Certificates | SSH` *section*.

**Note**

openmediavault will require the host key of the remote server prior to running any job with a remote server. This can be accomplished in a root SSH session by attempting to log into the server, even if the root user does not have a public key on the remote server:

```
# ssh example.com
The authenticity of host 'example.com (203.0.113.8)' can't be established.
ED25519 key fingerprint is SHA256:WohJIIwTqt5aL4ne83TkPEn3wZ9zMWiRvY82Gw93JR8.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'example.com (ED25519)' to the list of known hosts
root@example.com's password:
```

These are the options most commonly used in RSync. At the end there is an extra text field where you can add more options.

**Configuration**

openmediavault makes the tasks run by placing them in `/etc/cron.d/openmediavault-rsync` one line per job. The cron time at the beginning, then user (root) and the target file that holds the actual RSync file with the final command, is configured in the same way as *scheduled tasks* The files are stored in `/var/lib/openmediavault/cron.d/`, prefixed with `rsync` and a <uuid>. A default SSH RSync job looks like this:

```
#!/bin/sh
# This configuration file is auto-generated.
# WARNING: Do not edit this file, your changes will be lost.
. /usr/share/openmediavault/scripts/helper-functions
cleanup() {
  omv_kill_children $$
  rm -f /var/run/rsync-05260f23-5098-4f07-9250-0b38b923ac7f
  exit
}
[ -e /var/run/rsync-05260f23-5098-4f07-9250-0b38b923ac7f ] && exit 1
if ! omv_is_mounted "/srv/dev-disk-by-label-VOLUME1/" ; then
    omv_error "Source storage device not mounted at </srv/dev-disk-by-label-VOLUME1>!"
    exit 1
fi
trap cleanup 0 1 2 5 15
touch /var/run/rsync-05260f23-5098-4f07-9250-0b38b923ac7f
omv_log "Please wait, syncing </srv/dev-disk-by-label-VOLUME1/backupdir/> to
↪<username@backupserver.com:/opt/backup> ...\n"
eval $(ssh-agent) >/dev/null
ssh-add /etc/ssh/openmediavault-484a6837-5170-468c-aa8f-0e3cb92a641e >/dev/null
rsync --verbose --log-file="/var/log/rsync.log" --rsh "ssh -p 22" --recursive --times --
↪archive --perms '/srv/dev-disk-by-label-VOLUME1/backupdir/' 'username@backupserver.
↪com:/opt/backup' & wait $!
omv_log "\nThe synchronisation has completed successfully."
```

### Server

This is the place for configuring the RSync daemon and its modules (shared folders).

**Settings**

Change listening port of the daemon and add extra configurations directives text field.

**Modules**

This is where you add shared folders to be available to the daemon. The options are explained in the module web panel. If you want to protect the modules you can select the next tab and choose a server username and establish a password. Be aware the password is only for the modules, is not the linux password. Documentation for the extra options for the modules is provided by rsyncd manual.

The above server settings are sent to this file `/etc/rsyncd.conf`.

### 5.3.6 Kubernetes

#### Overview

The Kubernetes service in openmediavault is designed to manage and orchestrate containerized applications using the Kubernetes platform. It provides a user-friendly interface to deploy and manage applications in a cluster environment.

The Kubernetes service can be installed via the *openmediavault-k8s* plugin.

The lightweight K3s is used under the hood, allowing a Kubernetes environment to run on SBC systems as well.

The installed applications can then be accessed via the URL:

```
https://<APPNAME>.<FQDN>:8443
```

Example:

```
https://whoami.mynas.internal:8443
```

**Note**

It is important that the local DNS server/router can resolve these hostnames.

#### Networking and DNS

By default, the Traefik reverse proxy in the openmediavault Kubernetes plugin uses subdomain-based URLs in the form `http://<APPNAME>.<FQDN>:8080` or `https://<APPNAME>.<FQDN>:8443`.

If local DNS cannot resolve these names, an alternative is to use sslip.io, which is a public DNS service that maps hostnames containing an IP address directly to that IP. This is useful when your local DNS server does not provide records for app subdomains.

Pattern:

```
https://<APPNAME>.<IP>.sslip.io:8443
```

Example:

```
https://whoami.192.168.1.10.sslip.io:8443
```

**Hint**

Most recipes are already prepared for `sslip.io` usage. In many cases, you only need to uncomment the corresponding line in the recipe configuration section.

Using subdomains (e.g., *whoami.mynas.internal*) is preferred over path prefixes (e.g., *mynas.internal/app*) for accessing apps via the Traefik reverse proxy in the openmediavault Kubernetes plugin because:

- Cleaner and simpler routing – Subdomains keep Ingress rules modular and avoid complex path rewrites.
- Fewer compatibility issues – Many apps assume they run at / and break when served under a path prefix due to issues with relative URLs, static assets, or routing.
- Avoids URL rewriting problems – With path prefixes, you often need to rewrite URLs or modify the app, which can cause subtle bugs or failures.

- Better isolation and security – Subdomains help enforce security boundaries between apps.
- Easier scaling and maintenance – Apps on subdomains are more portable, scalable, and microservice-friendly.

### Recipes

The Kubernetes service offers several, so called, recipes to help you get started with installing popular applications.

The recipes are maintained in the openmediavault-k8s-recipes GitHub project and are offered by the plugin in a graphical overview for easy installation.

The recipes have an area at the top of the YAML code that must be filled in by the user. This configuration area contains information that is then used by the recipe to install the application on the given system.

Installed applications are not updated by openmediavault, this must be done by the user. In most cases, it is sufficient to first delete the image and then the pod of the application to be updated.

### Administration

openmediavault offers a rudimentary UI for managing simple tasks in the cluster, but it makes sense to use external software that specializes in this area. Examples include Rancher UI (to be installed using a recipe) or k9s.

### Customization

The Kubernetes service in openmediavault is can be customized via *environment variables* to adapt it to your personal needs.

| Environment variable | Example | Description |
|---|---|---|
| OMV_K8S_K3S_VER | `v1.33.5+k3s1` | Specifies the K3s version to be used. To force a reinstallation of K3s with the specified version, create the file `/var/lib/openmediavault/upgrade_k3s` before running **`omv-salt deploy run k3s`**. |

## 5.4 Users

The user management in openmediavault is provided by the user management of the operating system on which it is running.

However, openmediavault also maintains control over these users, **so management is a team effort** between the Debian operating system and the internal database of openmediavault.

Users that are managed via the web interface are so-called *non-system users*. This type of users are identified by their *UID*, which is in a specific range, usually between 1000 and 60000. The same applies to *non-system groups* that are managed via the web interface. They are identified by their *GID* which is usually in the range from 1000 to 60000. Check `/etc/login.defs` for more information.

The **Users** management section in the web interface is divided into three subsections: `Settings`, `Users` and `Groups`.

Users can log into the web interface to see their own profile information. The administrator can prohibit this behaviour for each user individually.

### 5.4.1 Settings

#### User home directory

Due to the nature explained, users are supposed to have their own private place for files that is called "*home*", depending on the type of service, automatically becomes a personal private shared location.

You can optionally select a shared folder as root for the *home* directories of all non-system users.

If *User home directory* is disabled and a new user is created, the following happens:

- No home directory will be created and assigned to the user

If *User home directory* is enabled and a new user is created, the following happens:

- A home directory will be created in the selected shared folder and assigned to the user
- The "skel" templates from Debian will applied to the new home directory

If *User home directory* is enabled, the following actions will be performed on existing users:

- The home directory path will be updated for all existing non-system users.
- The previous home directory content will NOT be moved to the new location. This has to be done manually.

If *User home directory* is disabled, the following actions will be performed on existing users:

- The home directory will be unset for all existing non-system users.
- The home directory content will NOT be deleted.

### 5.4.2 User

This page lists all non-system users and allows you to *Create* or *Edit* those users as well as their shared folder *Permissions*. There is also a special option that allows you to *Import* multiple users at once.

#### Create

This page is intended for creating a new user. The following form fields are available:

**Name**
: This must be only numbers and letters. Its the "username" of the login credentials and must be all lowercase to avoid confusion.

**Password**
: This field will provide the password of the user.

**Shell**
: The shell is only used for remote access to interact with the server. By default the form will offers **`/usr/bin/sh`** shell, but is recommended usage of the **`/usr/bin/nologin`** shell to prevent local and remote console access.

**Groups**
: This field allows to add or remove users from specific groups. Groups are the means of access for multiple users to multiple shared resources.

  Some groups only affect the system (as of Linux), others are specific to the openmediavault system. By default all users created using the web interface are added to the `users` group (`gid=100`).

**SSH public keys**
: Add or remove *public SSH keys* for granting remote access for users.

**Disallow account modification**
: Disallow the user to modify their own account information.

**Tags**
Specify tags to categorize the user.

#### Import

Designed for bulk user creation. The user data must be entered as CSV data. An example is prepared as a comment.

Those fields are the same as the form of the *Create* user page.

The field of *UID* must be numeric and must be in the range from 1000 to 60000 (check `/etc/login.defs` for more information).

Example data:

```
# <name>;<uid>;<tags>;<email>;<password>;<shell>;<group,group,...>;<disallowusermod>
user1;1001;user1;user1@myserver.com;password1;/bin/bash;sudo;1
user2;1002;user2;user2@my.com;password2;/bin/sh;;0
user3;1003;user3;user3@example.com;password3;/bin/false;;1
user4;1004;user4;user4@test.com;password4;;;1
```

#### Edit

Here you can modify the user information, the fields are the same as the form of the *Create* user page.

#### Permissions

All existing shared folder and the access rights of the user to be edited are displayed on this page. The following access rights are available:

- Read/Write
- Read-only
- No access

These settings are used by the services to configure the access rights for the users.

> **Note**
>
> Please note that these settings have no effect on file system permissions.

How the permissions are stored is described further down in the *shared folder* section.

### 5.4.3 Group

This page lists all non-system groups and allows you to *Create* or *Edit* those groups as well as their shared folder *Permissions*. There is also a special option that allows you to *Import* multiple groups at once.

#### Create

This page is intended for creating a new group. The following form fields are available:

**Name**
This must be only numbers and letters.

**Members**
This field allows to add or remove users for this group.

#### Import

Designed for bulk group creation. The group data must be entered as CSV data. An example is prepared as a comment.

Those fields are the same as the form of the *Create* group page.

The field of *GID* must be numeric and must be in the range from 1000 to 60000 (check `/etc/login.defs` for more information).

#### Edit

Here you can modify the group information, the fields are the same as the form of the *Create* group page.

#### Permissions

All existing shared folder and the access rights of the group to be edited are displayed on this page. The following access rights are available:

- Read/Write
- Read-only
- No access

These settings are used by the services to configure the access rights for the groups.

**Note**

Please note that these settings have no effect on file system permissions.

How the permissions are stored is described further down in the *shared folder* section.

### 5.4.4 Technical details

When a user is created openmediavault backend executes **useradd** in non-interactive mode with all the information passed from the form fields. This command is responsible for creating an entry in `/etc/passwd` and a hashed password in `/etc/shadow`.

The openmediavault backend monitors all database changes to users to allow other services to react to these changes. This ensures, for example, that the *Samba* user database is updated when a user password is changed.

**Attention**

- The user profile information (except password) is also stored in the internal openmediavault database, along with the public keys.
- A non-privileged user can become a web interface administrator by adding them to the `openmediavault-admin` group.

#### Manual management

If a user is created via the web interface, no corresponding group with the name of the user is created.

A user created in terminal by the **useradd** command will not be in the openmediavault internal database. This causes trouble with some services, by example *Samba*, as there is no user/password entry in the `tdbsam` database of *Samba*.

To synchronize users or groups that have not been created in the web interface, simply perform an *Edit* action and change the password or membership.

#### Shared home directories

If *User Home directory* is enabled and configured properly, then the home directories can be shared by some services as well, e.g. *Samba* and *FTP*.

## 5.5 Software & Update Management

### 5.5.1 Overview

openmediavault is a Debian based distribution. It uses APT to install packages. All standard Debian packages are upgraded using the official Debian mirrors. openmediavault packages are upgraded using the https://packages.openmediavault.org repository.

### 5.5.2 Update Management

Package updates will be displayed under `System | Update Management | Updates` in the web interface. There you can view the changelog of individual packages or install all packages at once.

The software `cron-apt` is used by openmediavault to perform a daily scheduled job to check for new package updates. Security updates will be installed automatically in the background. If you have notifications enabled you will receive an email every time packages are ready for installation.

### 5.5.3 Using CLI

**omv-upgrade**

This is non-interactive wrapper script that basically re-synchronizes the package index files from their sources and installs the newest versions of all packages currently installed on the system.

**omv-release-upgrade**

This is a script which is included only in the last versions of openmediavault before moving to the next major release version, e.g. 5.6.x to 6.x. This command migrates the system to the next major openmediavault version (which mostly includes a Debian distribution upgrade as well).

### 5.5.4 Installing plugins

Plugins can be installed via the web interface or CLI. If the plugin requires some extra software, the Debian package management will fetch all dependencies from the configured Debian package repositories.

### 5.5.5 Installing Software

You can install all software available in the Debian repositories on your server.

> **Warning**
>
> Please note that the installation of additional software may impair or prevent the functionality of openmediavault. This includes especially the installation of newer Python package versions via PIP which have already been installed via APT. Also, please do not install packages from Debian `Backports`, `Testing` or `Experimental` repositories, as this may install incompatible package versions. openmediavault is strictly tied to the Debian version it is based on.
>
> Better use a container based solution to install additional software to do not affect the openmediavault operating system.

**Warning**

Do not install the `apache` webserver package, this leads to the uninstallation of the `openmediavault` package and the unusability of the system.

**Install**:

```
$ apt-get install <packagename>
```

**Remove**:

```
$ apt-get remove <packagename>
```

**Purge (remove package and configuration files)**:

```
$ apt-get purge <packagename>
```

### 5.5.6 Repositories

**Debian**

The OS repositories are in this file `/etc/apt/sources.list`.

**External**

Debian provides the `/etc/apt/sources.list.d/` folder for adding external repositories. The configuration files for the openmediavault package repositories are located here.

**Warning**

If there is a strong need to add the `Backports`, `Testing` or `Experimental` Debian repository just to install the most recent software, then please ensure that these packages are properly pinned[1] to avoid the system becoming unstable for adding core unsupported software by mistake.

## 5.6 Custom Configuration

openmediavault is not a replacement for webmin, where you can configure all options in the web interface. Options are already preconfigured to make it easier for the average user to install and start using the NAS server.

As mentioned before in the *FAQ* openmediavault takes full control of some services, making it difficult to intervene configuration files. Changes manually added to configuration files will eventually overwritten at some stage by the openmediavault system.

To overcome this there are some options available to modify some of the default openmediavault configuration options and values, like the use of environmental variables.

### 5.6.1 Environmental Variables

The web interface does not provide access to ALL the configuration aspects of a complex system like openmediavault. However, the system allows to change some advanced settings through the use of environment variables.

[1] https://wiki.debian.org/AptPreferences

#### List environment variables

A list of available environment variables can be collected via:

```
# omv-env list
```

To get a list of all configured environment variables use the following command:

```
# omv-env get
```

To get the value of a specific environment variable use:

```
# omv-env get <VARIABLE_NAME>
```

The list of environmental variables used for `/etc/fstab` filesystem with the defaults options and how to use them is described here.

#### Modify environment variables

To set or change these variables, run the following command:

```
# omv-env set -- <VARIABLE_NAME> <VALUE>
# omv-env set -- OMV_SSHD_SUBSYSTEM_SFTP "/usr/lib/openssh/sftp-server"
```

The configured environment variables are located in the file `/etc/default/openmediavault`.

#### Apply changes

To apply the custom settings you need to execute the following commands as root:

```
# monit restart omv-engined
# omv-salt stage run prepare
# omv-salt stage run deploy
```

### 5.6.2 The SaltStack states

If you want to deploy custom configuration settings, then you could add additional Salt states. Please check out the SaltStack documentation for more information about how it works and how to use it.

The openmediavault SLS files are located at `/srv/salt/omv/`.

#### Add custom states

If you want to customize the configuration of a service or any other application that is managed by openmediavault, then you need to know where to add your custom state file first. Start searching the location on GitHub. You will find the corresponding location below `/srv/salt/omv/deploy` in the root file system of your system. If there are no files starting with a number, e.g. `90cron.sls`, then this service is not customizable. It is still possible, but is beyond the scope of this introduction. The states are executed in ascending order.

Choose the order when your state needs to be executed and create a file like `<N>xxxx.sls`, where N is between 0 and 100.

To append something to an already existing configuration file use this YAML:

```
<UNIQUE_IDENTIFIER>:
  file.append:
    - name: "<PATH_OF_THE_FILE>"
```

(continues on next page)

(continued from previous page)

```
    - text: |
        <LINE1>
        <LINE2>
        <...>
```

Example:

```
customize_postfix_main:
  file.append:
    - name: "/etc/postfix/main.cf"
    - text: |
        mynetworks = 127.0.0.0/8 168.100.189.0/28
```

For more file modifications please have a look into the file module.

Finally you need to *deploy* your changes by running:

```
# omv-salt deploy run <SERVICE_NAME>
```

CHAPTER

SIX

# PLUGINS

You can add more features & apps by simply selecting the software you need, we call this plugins. Plugins are possible due to the modular design of openmediavault and are the preferred way to extend your NAS. While it is still possible to install regular software alongside openmediavault, installing instead containerized software build with podman or Docker is highly recommended. This recommendation stems from many issues caused by later version updates that made openmediavault or other software unusable due to introducing conflicting component dependencies. Plugins only exist for your convenience.

## 6.1 Benefits

Compared to adding regular software, plugins offer the following benefits:

- Easier to install - You just click on what you want.
- Easier to configure - it is often preconfigured so you don't have to.
- Automatic updates - ensure Stability & Security.
- A Webinterface - is added when needed for your ease of use.

## 6.2 Overview

The following is the list of official plugins by openmediavault. Their availability depends on the architecture on which openmediavault is installed.

- **K8s**: Run your Docker containers in a lightweight Kubernetes environment.
- **MD**: Create, manage, and monitor software RAIDs based on Linux MD (Multiple Device) devices.
- **ClamAV**: Provides Clam AntiVirus (ClamAV). It is a free software, cross-platform and open-source antivirus software toolkit able to detect many types of malicious software, including viruses
- **Diskstats**: Complementary plugin to extend system statistics collection by adding I/O statistic graphs.
- **Forked-daap**: Provides a daap protocol music server.
- **FTP**: Provides a modular FTP/FTPS server.
- **LVM2**: LVM managing. Create volume groups and logical partitions.
- **NUT**: Controlling and configuring UPS. The driver support is based on NUT.
- **Onedrive** (amd64, arm64, armhf, i386 only): Synchronizing a shared folder with Microsoft OneDrive cloud storage.
- **Podman**: A tool for managing containers and images, volumes mounted into those containers, and pods made from groups of containers. Gives users the ability to pull the latest images of the containers that are used by some of the plugins

- **Shairport**: Provides Airtunes emulator. Stream music wirelessly to your iPod/iPad/iPhone/iTunes.
- **ShareRootFs**: Provides shared directories on root file system.
- **SNMP**: Provides Simple Network Management Protocol (SNMP). SNMP is an Internet Standard protocol for collecting and organizing information about managed devices on IP networks.
- **TFtp**: Provides Trivial File Transfer Protocol (TFTP). TFTP is a simple lockstep File Transfer Protocol which allows a client to get a file from or put a file onto a remote host.
- **USB Backup**: Backup internal data to external disks on scheduled basis or on plug drive event.
- **BCache**: Use an SSD as a read/write cache for another block device.

Podman container-based plugins:

- **FileBrowser**: File managing interface.
- **OwnTone**: Provides a DAAP (iTunes), MPD (Music Player Daemon) and RSP (Roku) media server.
- **PhotoPrism**: AI-powered app for browsing, organizing & sharing your photo collection.
- **S3**: MinIO based high-performance, S3 compatible object storage.
- **WeTTY**: Terminal access in browser over HTTP/HTTPS.

## 6.3 3rd party

An overview of the third party plugin list can be found at omv-extras.org.

CHAPTER

SEVEN

# FAQ

## 7.1 What is OMV?

OMV is an abbreviation of openmediavault.

## 7.2 Is openmediavault a fork of FreeNAS?

No

## 7.3 When shall i use openmediavault?

If you want to set up a NAS really fast and manage it easily, then openmediavault is the right solution for you. If you want to use a really free software where you are not a beta tester for a commercial variant, then openmediavault is also a solution for you.

## 7.4 Does openmediavault have drivers for my hardware?

Please consult the Debian hardware compatibility list for more information.

## 7.5 Can I use a usb flash drive (stick) for installing the system?

Yes, but the installation does not have any optimizations to reduce writes into the OS disk. The usb media will most likely start failing within a few weeks of usage. Most common symptom is basic command execution does not work, denied login, etc. More information can be found here.

## 7.6 What are the default credentials for the UI?

Use the user 'admin' and the password 'openmediavault' for the first login.

## 7.7 Can I give administrator privileges to non-privileged users to access the web control panel?

Yes. By default non-privileged users can only access their account profile, they can change password and their email address if the administrator has allowed changes on their account. However the current web interface framework is designed for developers to create plugins where they can give limited or full access to non-privileged users. An example is the openvpn plugin by omv-extras.org. A non-privileged user can become a web interface administrator by adding them to the `openmediavault-admin` group.

## 7.8 I cannot read out SMART data for my storage device?

For storage devices connected via USB, openmediavault relies entirely on the auto-detection functionality of the smartmontools applications. To be always up to date openmediavault automatically downloads the smartctl/smartd database once a week. You can do that manually by running **update-smart-drivedb** in the CLI. Should it nevertheless happen that a storage device is not supported, please submit a request to the smartmontools project. You can find more information in their FAQ. If you can not wait till the upstream project has integrated your device, then you can add it manually to a local database at */etc/smart_drivedb.h*. Please check their FAQ for more information.

## 7.9 What is the file `/etc/openmediavault/config.xml` for?

It is the database configuration store file for openmediavault. When a change is performed in the web interface, the config value is stored and/or retrieved by RPC to/from this file.

## 7.10 Can I upgrade to Debian Testing/Unstable (Debian Testing/Sid) or use Ubuntu as a base distribution?

Yes. But the end is most likely a broken web interface and possibly broken system. openmediavault releases are heavily tight to their Debian base distribution. Get more information *here*.

## 7.11 I´ve lost the web interface password. How do I reset it?

Simply connect via ssh into the server or login locally on the machine and type in: **omv-firstaid**. There is an option to reset the web interface password.

## 7.12 Can I backup or restore an existing openmediavault configuration?

There is no regular backup/restore procedure, but yes, in some way: keep the file `/etc/openmediavault/config.xml` for references purposes if the option is to go for a clean re-install.

## 7.13 What is the default HTTP engine of openmediavault?

NGINX. The last version of openmediavault with Apache was 0.5 Sardoukar.

## 7.14 Can I use Apache as HTTP engine?

Yes, but is not supported. Eventually every openmediavault package upgrade will activate NGINX again leaving the web interface broken. A parallel Apache instance next to Nginx is possible, just make sure the ports are different otherwise the openmediavault web interface will not work.

## 7.15 How can use the default HTTP engine to hold my own web page?

Do not modify openmediavault default NGINX files. Place the website configurations in `/etc/nginx/sites-available` and enable it with **nginx_ensite <SITE>**. Read more information in the NGINX documentation.

## 7.16 Why does the system rewrites a configuration file(s) that I have manually edited?

OMV takes full control of some system services. This services include monit, ntp, samba, network, proftpd, nginx, php5-fpm, etc. Read here.

## 7.17 How can I modify an internal value of some service openmediavault has control over?

Read *here* for advanced configurations.

## 7.18 How can I modify or add a network configuration with some custom options the web interface does not provide?

Starting with openmediavault version 5 *systemd-networkd* is used to configure the network. The interfaces file `/etc/network/interfaces` is controlled by openmediavault but not used anymore. To add network interfaces that are not configurable through the web interface or other options not present, use *advanced settings*. Alternatively write your own *systemd-networkd* configuration files.

## 7.19 Why my disks mount paths have a long alphanumeric number?

The long number is called UUID, it is used by fstab to mount disks. This number is unique per filesystem (or at least unlikely possible that another filesystem comes with an identical one). This helps maintaining the mount points. The old linux way (sda1, sdb1, etc.) is not guaranteed that /dev/sda1 is the same disk on next reboot. If having trouble identifying them in terminal, create a pool with symlinks to each file system with easy to remember names.

This behaviour has been deprecated now in current openmediavault releases. The default creation of mount paths is documented here.

## 7.20 I don't have a data disk, and I want to use my OS disk for storing data?

The default behaviour of openmediavault is to act as NAS server, that means OS files are separated from data disks.

However if the OS disk is partitioned the system will recognise the extra partitions besides rootfs if is formatted. You can mount it and use it for shared folders.

The current installer does not provide access to the partition manager, use a plain Debian iso then install openmediavault on top and accommodate the partitions, or resize the partition after installing using Gparted or SystemRescueCd.

## 7.21 Can I install openmediavault on top a running Debian system?

Yes, but it is recommended that the current running OS not to have a desktop environment installed.

## 7.22 What is the permissions/ownership of folders in openmediavault created by shared folders?

The default is folders in `2775` mode, with `root:users` ownership. This means all users created in the web interface can read, write to folders created by the system in the data drives using the default. The setgid allows group inheritance, meaning new files/folders below will always have the group users (GID=100) membership.

## 7.23 I need to delete a shared folder, why the delete button is greyed/disabled?

Shared folder configurations can be used across different services. When removing a shared folder configuration is necessary to unlink it from every service is attached to, before the delete button becomes available. At the moment there is no internal database backend that can display information about which service is holding which shares.

## 7.24 What is the `omv-salt` command for?

**`omv-salt`** is a terminal console command that is used by the backend of openmediavault to pipe directives and values to service configuration files. The arguments that **`omv-salt`** accepts are related to the name of the service it configures. Type **`omv-salt`** in terminal, press TAB key, and the terminal will display all available arguments.

## 7.25 I want to experiment with openmediavault or make changes to the code.

As a true open source system everything is possible. The recommendation is do not test with the production server to avoid breaking the web interface. The best thing to do is to use a Virtual Machine. On Sourceforge there are preconfigured openmediavault images with virtual disks ready to launch. Alternatively checkout the openmediavault GIT repository and use Vagrant to create a virtual machine.

## 7.26 What is the `omv-upgrade` and `omv-release-upgrade` for?

Information about those commands are in the software *section*.

CHAPTER

EIGHT

# TROUBLESHOOTING

## 8.1 The web interface has missing fields and/or items showing that have been uninstalled.

Clear your browser cache.

## 8.2 The web interface keeps rejecting my credentials.

If the password is correct then this is most likely caused by the rootfs partition being full. This command can help track which folders are the biggest `df -hx --max-depth=1 /`.

## 8.3 I only see a few items in the web interface interface like the user section of Access Rights Management.

You did not login as the admin user. This is the only user that can access everything.

## 8.4 I get an error every time I post in the forum especially if it is a long post and/or has links to external pages.

The error is deceiving. Please don't keep trying to post. The spam filter has flagged your post and it will need to be approved. Please be patient.

## 8.5 I mounted the drive using the command line or GUI tool and I can't pick that drive in the shared folder device dropdown.

Never mount a drive with anything other than the openmediavault web interface. This creates the necessary database entries to populate the device dropdown.

## 8.6 Samba is slow.

Read these threads:

- read/write performance for SMB shares hosted on RPi4
- Tuning Samba for more speed (note: written for OMV 2.x)
- Tuning Samba for more speed 2 (note: written for OMV 2.x)

## 8.7 Samba share password is refused from Windows 10.

To fix the problem you need to change the Network Security LAN Manager authentication level.

## 8.8 How to troubleshoot an error caused by an "Option" parameter passed to a plug-in?

To find the root cause, run the faulty systemd unit file yourself by executing:

```
systemd restart <plug-in-daemon>
```

If output of <plug-in> is now more verbose, then you will get a hint on STDOUT. If not, then you need to run `journalctl -f` in parallel to get the syslog output. Admittedly, not really novice friendly, but it's really not possible to do it any other way. OMV always tries to be as error/debug friendly as possible; by default.

## 8.9 I am using JMicron drive enclosures and some of my drives are not appearing.

This is likely because JMicron controllers incorrectly report identical serial numbers and other data which confuses various systems. openmediavault provides an UDEV rules database which will fix that issue for several USB PATA/SATA bridge controllers. If your hardware still does not work, then please provide the information mentioned in that pull request and open a new tracker issue.

Alternatively you can manually "fix" this by adding a rule to `/lib/udev/rules.d/60-persistent-storage.rules` after the entry for *Fall back usb_id for USB devices*:

```
# JMicron drive fix
KERNEL=="sd*", ENV{ID_VENDOR}=="JMicron", SUBSYSTEMS=="usb", PROGRAM="serial_
↪id %N", ENV{ID_SERIAL}="USB-%c", ENV{ID_SERIAL_SHORT}="%c"
```

This will ensure that unique paths are created based on the serial number of the actual drives and not the enclosures.

CHAPTER

NINE

# DEVELOPMENT

## 9.1 Coding Guideline

These standards for code formatting and documentation must be followed by anyone contributing to the openmediavault project. Any contributions that do not fullfill these guidelines will not be accepted.

### 9.1.1 File Formatting

**Indentation**
Use 4 space tabs for writing your code. If you are modifying someone else's code, try to keep the coding style similar.

**Line Length**
Lines shouldn't be longer than 80 characters.

**Line Endings**
Line endings should be Unix-style LF.

**Encoding**
Files should be saved with UTF-8 encoding.

### 9.1.2 Naming Conventions

**Classes**
PHP classes should use the OMV namespace.

```
namespace OMV\System;

/**
 * @ingroup api
 */
class Process {
        use \OMV\DebugTrait;
    ...
```

**Functions/Methods**
Functions/Methods must use camel case syntax, this convention capitalizes the first character of each word except the first one.

```
public function getGecos() {
...
}
```

(continues on next page)

(continued from previous page)

```
public function getHomeDirectory() {
...
}
```

**Variables**

Variables must use camel case syntax, this convention capitalizes the first character of each word except the first one.

```
$fsName
$outputFileName
```

**Constants**

Constants should start with OMV_ and should be all upper case.

```
$OMV_DEFAULT_FILE = "/etc/default/openmediavault";
$OMV_JSONSCHEMA_SORTFIELD = '"type":["string","null"]';
```

### 9.1.3 Multiline parameters

Functions with many parameters may need to be split onto several lines to keep the 80 characters/line limit. The first parameters may be put onto the same line as the function name if there is enough space. Subsequent parameters on following lines are to be indented using 1 tab.

```
throw new OMVException(OMVErrorMsg::E_EXEC_FAILED,
    $cmd, implode("\n", $output));

$dispatcher->notify(($data['uuid'] == $GLOBALS['OMV_UUID_UNDEFINED']) ?
    OMV_NOTIFY_CREATE : OMV_NOTIFY_MODIFY,
    "org.openmediavault.system.storage.hdparm", $object);
```

### 9.1.4 Control Structures

```
for (i = 0; i < 10; i++) {
    if (foo(i)) {
        bar();
    }
}

switch (x) {
case 'a':
    ...
    break;
case "b":
    ...
    break;
default:
    ...
    break;
}

if (a) {
    foo();
```

(continues on next page)

(continued from previous page)

```
} else {
    bar();
}

if (TRUE === $result)
    break;

foreach ($output as $outputk => $outputv) {
    foo();
}
```

### 9.1.5 Comments

**Single line comments**

You should use the // comment style to "comment out" code. It may be used for commenting sections of code too. Single line comments must be indented to the indent level when they are used for code documentation.

**Block comments**

Block comments should usually be avoided. For descriptions use the // comments.

```
// Parse output:
// shadow:x:42:openmediavault
// snmp:x:112:
// sambashare:x:113:
// openmediavault:x:999:
// nut:x:114:
foreach ($output as $outputv) {
    ...
}
```

**Documentation comments**

Use the doxygen syntax where possible.

```
/**
 * Get the filesystem label.
 * @return string The filesystem label, otherwise FALSE.
 */
public function getLabel() {
    ...
}

/**
 * Enumerate all disk devices on the system.
 * @return array An array containing physical disk device objects with
 *   the fields \em devicename, \em devicefile, \em model, \em size,
 *   \em description, \em vendor, \em serialnumber, \em israid and
 *   \em isrootdevice.
 */
public function enumerateDevices() {
    ...
}
```

(continues on next page)

(continued from previous page)

```
/**
 * Enumerate all disk devices on the system. The field \em hdparm will
 * be added to the hard disk objects if there exists additional hard
 * disk parameters (e.g. S.M.A.R.T. or AAM) that can be defined
 * individually per hard disk.
 * @param array data An array containing the following fields:
 *   \em start The index where to start.
 *   \em limit The number of objects to process.
 *   \em sortfield The name of the column used to sort.
 *   \em sortdir The sort direction, ASC or DESC.
 * @return array An array containing the requested objects. The field
 *   \em total contains the total number of objects, \em data contains
 *   the object array. An exception will be thrown in case of an error.
 */
public function getList($data) {
    ...
}
```

## 9.2 Contribute

If you want to contribute to the openmediavault project you have to subscribe a Contributor License Agreement.

The CLA used by openmediavault is a copy of the one used by the Apache Software Foundation for all contributions to their projects. This particular agreement has been used by other software projects in addition to Apache and is generally accepted as reasonable within the Open Source community.

You can easily subscribe the CLA with a single mouse click when submitting a pull request on GitHub.

Note that the CLA is not a transfer of copyright ownership, this simply is a license agreement for contributions. You also do not change your rights to use your own contributions for any other purpose.

### 9.2.1 Why do i have to subscribe a CLA when contributing to the openmediavault project?

- http://oss-watch.ac.uk/resources/cla
- http://www.golem.de/1107/85208.html
- Rechtliche Sicherheit bei Open-Source-Beiträgen
- Django's CLA FAQ
- Karl Fogel's Producing Open Source Software on CLAs
- The Wikipedia article on CLAs
- Gemeinsame Durchsetzung der Rechte

### 9.2.2 Can I submit patches without having signed the CLA?

No. All contributors and patch submitters must sign the CLA before they submit anything substantial. Trivial patches like spelling fixes or missing words won't require an agreement, since anybody could do those. However, almost anything will require a CLA.

### 9.2.3 Contribution rules

Code contributions are welcome, please read the guideline how to do this.

### 9.2.4 Anything else?

First time contributors should have a look into this talk.

Other useful information:

- How to write a Git commit message

### 9.2.5 How to become a translator?

If you want to help translating the openmediavault web interface please do the following:

- Subscribe the CLA and send it to the given email address.
- Create an account at Transifex and join the openmediavault project as translator.
- You will get notified when your request has been approved. You will be listed as contributor here.

## 9.3 How does it work?

openmediavault is a complex piece of software. Developers can easily understand how it works by looking at the source code and developing plugins. For the average user it is a little bit more complicated. Some of the mechanics on how it works are explained ahead.

**Essential elements:**

**Frontend** The web interface, is written in TypeScript using the Angular and Angular Material frameworks. The page is delivered using the nginx web server.

**Backend** PHP code, executes several tasks.

**config.xml** Database file in XML format, located in `/etc/openmediavault` We will refer in this explanation just as config.xml

**omv-salt** Tool to *deploy* the configuration and services.

**omv-engined** RPC daemon that runs all the PHP backend code. The nginx web server connects to this daemon through the FastCGI PHP socket. If an error appears in the web interface that indicates no connection to the PHP socket means the daemon is not running.

**Listeners** PHP backend code that listens to changes in the database. They are located in the modules section of the code `/usr/share/openmediavault/engined/modules`.

**/var/lib/openmediavault/dirtymodules.json** File that enumerates all sections that need to be reconfigured after the database has been written.

### 9.3.1 web interface values to/from the database

#### Read

When a user clicks on a panel, the JavaScript code retrieves values from config.xml using a RPC call. The RPC call has two main components usually the service name of the section (is in the js code) and the method. Press F12 in your favorite browser to open the developer console and go to Samba section in the web interface. In the browsers developer tools network section you will see several rpc.php calls. Find the one related to Samba, right click and select copy as cURL.

This is the json payload sent to omv-engined:

```
{
   "service":"SMB",
   "method":"getSettings",
   "params":null,
   "options":null
}
```

and the response:

```
{
  "response":{
     "enable":true,
     "workgroup":"HOME",
     "serverstring":"%h server",
     "loglevel":0,
     "usesendfile":true,
     "aio":true,
     "nullpasswords":false,
     "localmaster":false,
     "timeserver":false,
     "winssupport":false,
     "winsserver":"",
     "homesenable":false,
     "homesbrowseable":true,
     "extraoptions":""
  },
  "error":null
}
```

Just after the one before, another call, this time to get the Samba share list:

```
{
   "service":"SMB",
   "method":"getShareList",
   "params":{
      "start":0,
      "limit":25,
      "sortfield":"sharedfoldername",
      "sortdir":"ASC"
   },
   "options":null
}
```

And the response:

```
{
   "response":{
      "total":1,
      "data":[
         {
            "uuid":"9e4c8405-b01c-40b6-8c46-af6be17a1ff6",
            "enable":true,
            "sharedfolderref":"7ee2e4d0-8173-442b-88b9-63b4c731f920",
```

(continues on next page)

(continued from previous page)


```
            "comment":"",
            "guest":"no",
            "readonly":true,
            "browseable":true,
            "recyclebin":false,
            "recyclemaxsize":0,
            "recyclemaxage":0,
            "hidedotfiles":true,
            "inheritacls":true,
            "inheritpermissions":false,
            "easupport":false,
            "storedosattributes":false,
            "hostsallow":"",
            "hostsdeny":"",
            "audit":false,
            "extraoptions":"",
            "sharedfoldername":"sf1"
         }
      ]
   },
   "error":null
}
```


### Write

A user can do a simple task as to create a shared folder or change some settings in a service section. Whenever the user hits the save button, all fields from the section are submitted from the frontend via RPC to the internal database in `config.xml`, even the ones that are not changed. This is similar on what happens when reading values however the method here is named differently when saving: `setSettings`.

Stopping here, examining `config.xml` in terminal will see all the new stored values, what follows is that usually a yellow notification bar will appear in the web interface to indicate that it is necessary to apply changes. The yellow notification bar happens for one reason only: the `dirtymodules.json` file.

So the save button does two things actually, sends information to `config.xml` and what is called mark the relevant module as dirty. As en example: Making a change in general Samba or its shares will create a `dirtymodules.json` file like this:


```
[
    "samba",
    "zeroconf"
]
```


### Reconfiguring services

When the apply button is pressed, this very long PHP function gets executed.

In the following order, this will happen in background:

**`omv-salt deploy run samba`** -> `/etc/samba/smb.conf` will be completely rewritten –> Samba daemon is restarted

**`omv-salt deploy run zeroconf`** –> All files at `/etc/avahi/services/`*`ftp`*`,`*`smb`*`,`*`web`*`,`*`ssh`*`,`*`nfs`*`.service` will be rewritten –> Avahi daemon is restarted

That PHP function also performs checks for dependencies, in case a configuration needs to be reconfigured or reloaded before/after another one.

**Why is Zeroconf marked dirty?**

Because the Samba openmediavault code indicates that whenever a change is performed in this section, Zeroconf must be marked dirty. This is by design, Avahi is configured to announce Samba server if is enabled, so needs to know if openmediavault Samba server is enabled or disabled. If the database shows it is disabled the Avahi service file will be removed. The module backend is something all plugins can use. For example, a plugin that wants to use the privilege database model will have to listen to any changes in the shared folder database so it can reconfigure its files accordingly.

**What can break the web interface?**

As explained, the web interface depends on several third party software components.

1 - Nginx HTTP engine. The web server software is very sensitive to any syntax mistakes in `sites-available` folder. Any files there that do not pass syntax check will result in a fail to restart/reload nginx daemon. Also editing the openmediavault-webui nginx file improperly will result in failure. Nothing will be displayed by the browser, it will just say "Connection refused", as there is no software running on the HTTP port.

2 - omv-engined not running. Whenever the RPC daemon is not running, an error will pop in web interface "Failed to connect to socket: No such file or directory".

3 - The php-fpm socket is not running. Uncommon error, but if fiddling around with the PHP socket configuration or systemd to make it not start the web interface will display "502 Bad gateway".

All of the above errors should be able to be corrected with *omv-firstaid*. Offending files in sites-available should be removed from there to start the nginx server.

**Note**

As noticed how openmediavault works, the software does not parses configuration files. Any changes users add manually to smb.conf or any other configuration file will not be reflected in the web interface. This is why some hardcoded values are suggested to be customized via environmental variables. It can happen that a plugin marks Samba as dirty by design then the apply button will rewrite everything and restart it also.

Not every component in openmediavault is executed in the way described above. For example the filesystem backend has a much more complex mechanism.

## 9.4 Internal Tools

openmediavault software comes with several command line tools that can be used by developers and/or advanced users. Also it can be used to gather support information.

### 9.4.1 omv-salt

This tool is used to deploy the configuration of services and to start or stop them.

To get a list of all available deployment states:

```
# omv-salt deploy list
```

To get a list of all states that are dirty (saved in the Web UI but not yet reconfigured):

```
# omv-salt deploy list-dirty
```

To deploy one or more states run the following command:

```
# omv-salt deploy run <NAMES>...
# omv-salt deploy run avahi monit systemd
```

The tool also supports stages that bundle various tasks. To get a list of them, run the command:

```
# omv-salt stage list
```

The following stages are available:

- **setup**
  This stage is being run only once after the openmediavault Debian package has been installed on the system. It is used to set up the system to the desired requirements.
- **prepare**
  This stage takes care that the pillar and grains are up to date and all modules/states are being synced to the minions.
- **deploy**
  Deploy the configuration of various services like SMB, FTP, ...
- **all**
  This stage contains the stages prepare and deploy.

If you want to deploy all states in one bunch, then you need to execute the following command:

```
# omv-salt stage run deploy
```

If you want to deploy only the states that are dirty, run the command:

# omv-salt deploy run –append-dirty

### 9.4.2 omv-confdbadm

Most users tend to access/modify the database by using nano:

```
$ nano /etc/openmediavault/config.xml
```

This is a problem as sometimes a wrong pressed key can add strange chars out of the xml tags and make the database unreadable by the backend.

**omv-confdbadm** is a tool written in python for retrieving, storing or deleting values from/to the database. This tool combined with **jq**[1] provides an easier method for interacting with the database using Shell/BASH.

To read values in the database the tool needs as last argument the datamodel path. You can find all data models path here `/usr/share/openmediavault/datamodels/` prefixed with conf. Or list them with **omv-confdbadm list-ids**

Lets read all the registered filesystems that have been mounted through the web interface. Type the following command as root:

```
# omv-confdbadm read --prettify conf.system.filesystem.mountpoint
```

Output returns:

```
[
    {
        "dir": "/srv/dev-disk-by-label-ironwolf_3TB_1",
        "freq": 0,
```

(continues on next page)

[1] https://stedolan.github.io/jq/manual/v1.5/

(continued from previous page)


```
        "fsname": "/dev/disk/by-label/ironwolf_3TB_1",
        "hidden": false,
        "opts": "defaults,noauto,user_xattr,usrjquota=aquota.user,grpjquota=aquota.group,
↪jqfmt=vfsv0,acl",
        "passno": 2,
        "type": "ext4",
        "uuid": "567c2bd4-3d82-45b2-b34b-a6d38e680ed3"
    },
    {
        "dir": "/media/a448c5e9-7a50-4d48-b73d-48cadbe0326e",
        "freq": 0,
        "fsname": "a448c5e9-7a50-4d48-b73d-48cadbe0326e",
        "hidden": true,
        "opts": "noauto,",
        "passno": 0,
        "type": "fuse.mergerfs",
        "uuid": "4adf0892-cf63-466f-a5aa-80a152b8dea6"
    },
    {
        "dir": "/export/videos",
        "freq": 0,
        "fsname": "/media/a448c5e9-7a50-4d48-b73d-48cadbe0326e/data_general/videos",
        "hidden": false,
        "opts": "bind,noauto",
        "passno": 0,
        "type": "none",
        "uuid": "4457831c-309e-4693-8b0d-5db6b257194d"
    },
    {
        "dir": "/export/PVR",
        "freq": 0,
        "fsname": "/media/a448c5e9-7a50-4d48-b73d-48cadbe0326e/data_general/videos/pvr",
        "hidden": false,
        "opts": "bind,noauto",
        "passno": 0,
        "type": "none",
        "uuid": "dce89b85-f1e1-42e2-8d46-986de599abff"
    }
]
```


The first one is a native ext4 filesystem, the second object is storage pool, the last two are NFS binds.

**Filtering:** Get all filesystem mountpoints:

```
# omv-confdbadm read conf.system.filesystem.mountpoint  | jq -r '.[]|.dir'
```

Output returns:

```
/media/dev-disk-by-label-ironwolf_3TB_1
/media/a448c5e9-7a50-4d48-b73d-48cadbe0326e
/export/videos
/export/PVR
```

**Selecting:** Get all filesystem objects that are registered as ext4:

```
# omv-confdbadm read conf.system.filesystem.mountpoint  | jq -r '.[]|select(.type=="ext4")
↪'
```

Output returns:


```
{
  "opts": "defaults,noauto,user_xattr,usrjquota=aquota.user,grpjquota=aquota.group,
↪jqfmt=vfsv0,acl",
  "uuid": "567c2bd4-3d82-45b2-b34b-a6d38e680ed3",
  "passno": 2,
  "dir": "/media/dev-disk-by-label-ironwolf_3TB_1",
  "fsname": "/dev/disk/by-label/ironwolf_3TB_1",
  "freq": 0,
  "hidden": false,
  "type": "ext4"
}
```


**Write:** This tool can also modify values in the database.

Add the *noexec* flag to this filesystem object `567c2bd4-3d82-45b2-b34b-a6d38e680ed3`, we need to pass the whole json object as argument:

```
# omv-confdbadm update conf.system.filesystem.mountpoint '{"freq":0,"hidden":false,"passno
↪":2,"opts":"defaults,noexec,noauto,user_xattr,usrjquota=aquota.user,grpjquota=aquota.
↪group,jqfmt=vfsv0,acl","dir":"/media/dev-disk-by-label-ironwolf_3TB_1","uuid":
↪"567c2bd4-3d82-45b2-b34b-a6d38e680ed3","fsname":"/dev/disk/by-label/ironwolf_3TB_1",
↪"type":"ext4"}'
```

Remove a filesystem from the database, this time we pass only the corresponding uuid of the object:

```
# omv-confdbadm delete --uuid 567c2bd4-3d82-45b2-b34b-a6d38e680ed3 conf.system.
↪filesystem.mountpoint
```

### 9.4.3 omv-rpc

This tool can execute RPC commands. This is identical of what the web frontend uses to set/get information. It accepts service, method and parameters. RPC services can be found listed in engined/rpc folder

**Example 1:** Get all mounted filesystems, including rootfs:

```
# omv-rpc -u admin 'FileSystemMgmt' 'enumerateMountedFilesystems' '{"includeroot": true}'
```

Output returns:


```
[
 {
   "devicefile": "/dev/sda1",
   "parentdevicefile": "/dev/sda",
   "uuid": "752dee88-11a3-4524-848e-d50baf0211a2",
   "label": "",
   "type": "ext4",
   "blocks": "9738548",
   "mountpoint": "/",
   "used": "5.44 GiB",
```


(continues on next page)

(continued from previous page)


```
  "available": "3595554816",
  "size": "9972273152",
  "percentage": 62,
  "description": "/dev/sda1 (3.34 GiB available)",
  "propposixacl": true,
  "propquota": true,
  "propresize": true,
  "propfstab": true,
  "propcompress": false,
  "propautodefrag": false,
  "hasmultipledevices": false,
  "devicefiles": [
    "/dev/sda1"
  ]
},
{
  "devicefile": "dfa",
  "parentdevicefile": null,
  "uuid": null,
  "label": "dfa",
  "type": "zfs",
  "blocks": 901386.24,
  "mountpoint": "/dfa",
  "used": "5.26 MiB",
  "available": 917504000,
  "size": 923019509.76,
  "percentage": 0,
  "description": "dfa (875.00 MiB available)",
  "propposixacl": true,
  "propquota": false,
  "propresize": false,
  "propfstab": false,
  "propcompress": false,
  "propautodefrag": false,
  "hasmultipledevices": false,
  "devicefiles": "dfa"
},
{
  "devicefile": "/dev/sdg1",
  "parentdevicefile": "/dev/sdg",
  "uuid": "b50987a4-f111-4e94-a52e-9e6b204ac227",
  "label": "vol3",
  "type": "ext4",
  "blocks": "2030396",
  "mountpoint": "/srv/dev-disk-by-label-vol3",
  "used": "6.01 MiB",
  "available": "2056044544",
  "size": "2079125504",
  "percentage": 1,
  "description": "vol3 (1.91 GiB available)",
  "propposixacl": true,
  "propquota": true,
```


(continues on next page)

(continued from previous page)

```
   "propresize": true,
   "propfstab": true,
   "propcompress": false,
   "propautodefrag": false,
   "hasmultipledevices": false,
   "devicefiles": [
     "/dev/sdg1"
   ]
 }
]
```

**Example 2:** Get all block devices with no filesystem signatures. This is used by the RAID creation window:

```
# omv-rpc -u admin 'RaidMgmt' 'getCandidates' | jq
```

Output returns:

```
[
 {
   "devicefile": "/dev/mapper/vg-lv1",
   "size": "1296039936",
   "vendor": "",
   "serialnumber": "",
   "description": "LVM logical volume lv1 [/dev/mapper/vg-lv1, 1.20 GiB]"
 },
 {
   "devicefile": "/dev/mapper/vg-lv1",
   "size": "1296039936",
   "vendor": "",
   "serialnumber": "",
   "description": "LVM logical volume lv1 [/dev/mapper/vg-lv1, 1.20 GiB]"
 },
 {
   "devicefile": "/dev/sde",
   "size": "1610612736",
   "vendor": "QEMU",
   "serialnumber": "drive-scsi5",
   "description": "QEMU HARDDISK [/dev/sde, 1.50 GiB]"
 },
 {
   "devicefile": "/dev/sdf",
   "size": "2147483648",
   "vendor": "QEMU",
   "serialnumber": "drive-scsi4",
   "description": "QEMU HARDDISK [/dev/sdf, 2.00 GiB]"
 },
 {
   "devicefile": "/dev/sdj",
   "size": "1073741824",
   "vendor": "ATA",
   "serialnumber": "QM00009",
   "description": "QEMU HARDDISK [/dev/sdj, 1.00 GiB]"
```

(continues on next page)

(continued from previous page)

```
 }
]
```

The **jq** tool is used to prettify the output in JSON.

### 9.4.4 Shell helper-functions

openmediavault ships with this file `/usr/share/openmediavault/scripts/helper-functions` that contains several POSIX shell functions. To test them just run in terminal:

```
$ source /usr/share/openmediavault/scripts/helper-functions
```

Type `omv_`, press tab key to autocomplete, this will show all functions and a small description in the name.

**Example 1:** Shared folders objects in the database do not have their complete absolute path, it has to be constructed from the relative directory and the parent filesystem. If we know the shared folder database object <uuid> then:

```
$ omv_get_sharedfolder_path 2a8b04de-4e6c-4675-b761-1ddfabde2d2a
```

Returns:

```
/media/dev-disk-by-label-VOLUME1/Videos/Unsorted
```

**Example 2:** Database nodes need to be created when a plugin is installed and removed when it is purged. This is from omv-extras MiniDLNA plugin postinst file

```
omv_config_add_node "/config/services" "${SERVICE_XPATH_NAME}"
omv_config_add_key "${SERVICE_XPATH}" "enable" "0"
omv_config_add_key "${SERVICE_XPATH}" "name" "MiniDLNA Server on OpenMediaVault"
omv_config_add_key "${SERVICE_XPATH}" "port" "8200"
omv_config_add_key "${SERVICE_XPATH}" "strict" "0"
omv_config_add_key "${SERVICE_XPATH}" "tivo" "0"
omv_config_add_key "${SERVICE_XPATH}" "rootcontainer" "."
omv_config_add_node "${SERVICE_XPATH}" "shares"
omv_config_add_key "${SERVICE_XPATH}" "loglevel" "error"
omv_config_add_key "${SERVICE_XPATH}" "extraoptions" ""
```

Notice in the `postinst` file how it sources at the beginning `helper-functions`.

> **Note**
>
> What each function do and the parameters it accepts is documented in the helper-functions file .

## 9.5 Plugin Development

The range of functions of openmediavault can be expanded by the use of plugins. This section describes how to implement such plugins.

Plugins are implemented in a declarative manner, this means no JavaScript or TypeScript knowledge is needed.

### 9.5.1 Overview

A plugin contains of various YAML files that have to be located in the directory `/usr/share/openmediavault/workbench`. The following subdirectories each have a special meaning.

**component.d**
: This directory contains the manifest files of the pages shown in the openmediavault web interface.

**dashboard.d**
: This directory contains the manifest files of the dashboard widgets.

**log.d**
: This directory contains the manifest files that are used to configure the log content that is shown in the `Diagnostics | System Logs` datatable.

**navigation.d**
: This directory contains the manifest files that are used to configure the navigation bar on the left side of the web interface.

**route.d**
: This directory contains the manifest files that are used to configure the web interface routes.

A manifest file must follow the following schema:

```
version: "1.0"
type: component | dashboard-widget | log | navigation-item | route
data:
  ...
```

### 9.5.2 Directories

#### component.d

A manifest file must follow the following schema:

```
version: "1.0"
type: component
data:
  name: string
  type: navigationPage | formPage | selectionListPage | textPage | tabsPage |␣
↪datatablePage | rrdPage
  config:
    ...
```

The *name* field contains the unique identifier of the component. It is used to reference the component in a route configuration. The *type* field contains one of the following supported page types:

- blankPage
- codeEditorPage
- formPage
- selectionListPage
- textPage
- tabsPage
- datatablePage
- rrdPage

The available properties of each type can be found in the corresponding models.

Example:

```
version: "1.0"
type: component
data:
  name: omv-services-clamav-onaccess-scan-form-page
  type: formPage
  config:
    request:
      service: ClamAV
      get:
        method: getOnAccessPath
        params:
          uuid: "{{ _routeParams.uuid }}"
      post:
        method: setOnAccessPath
    fields:
      - type: confObjUuid
      - type: checkbox
        name: enable
        label: _("Enabled")
        value: false
      - type: sharedFolderSelect
        name: sharedfolderref
        label: _("Shared folder")
        hint: _("The location of the files to scan on-access.")
        validators:
          required: true
    buttons:
      - template: submit
        execute:
          type: url
          url: "/services/clamav/onaccess-scans"
      - template: cancel
        execute:
          type: url
          url: "/services/clamav/onaccess-scans"
```

### dashboard.d

The following dashboard widget types are available:

- grid
- datatable
- rrd
- chart
- text
- value

The available properties of each type can be found in the corresponding model.

Example:

```
version: "1.0"
type: dashboard-widget
data:
  id: 9984d6cc-741b-4fda-85bf-fc6471a61e97
  permissions:
    role:
      - admin
  title: _("CPU Usage")
  type: chart
  chart:
    type: gauge
    min: 0
    max: 100
    displayValue: true
    request:
      service: System
      method: getInformation
    label:
      formatter: template
      formatterConfig: "{{ value | tofixed(1) }}%"
    dataConfig:
      - label: Usage
        prop: cpuUsage
        backgroundColor: "#4cd964"
```

Check out the Dashboard widgets that are delivered with openmediavault for more examples.

### log.d

Plugins can add their own log files to the web interface. The properties of the manifest file can be inspected here.

Example:

```
version: "1.0"
type: log
data:
  id: clamav
  text: _("Antivirus")
  columns:
  - name: _("Date & Time")
    sortable: true
    prop: ts
    cellTemplateName: localeDateTime
    flexGrow: 1
  - name: _("Message")
    sortable: true
    prop: message
    flexGrow: 2
  request:
    service: LogFile
    method: getList
    params:
      id: clamav
```

#### navigation.d

To add a new item to the navigation bar on the left side of the web interface a manifest file with the following properties must be created.

The menu items are ordered alphabetically. If specified, the *position* field is added as additional sort condition.

Icons have to be specified like `mdi:<NAME>` or `<NAME>`. For the first format please have a look here for available icons. For the latter please check here. If possible, use the `<NAME>` format to ensure that uniform icons are used throughout the whole web interface.

Example:

```
version: "1.0"
type: navigation-item
data:
  path: "services.clamav.onaccess-scans"
  text: _("On Access Scans")
  position: 20
  icon: "mdi:file-eye"
  url: "/services/clamav/onaccess-scans"
```

#### route.d

A manifest file must follow the following schema:

```
version: "1.0"
type: route
data:
  url: string
  title: string
  editing: boolean
  notificationTitle: string
  component: string
```

The *url* is used to access the page via browser. A url like `/foo/bar` will finally look like `https://localhost/#/foo/bar`. The *title* field will be shown in the breadcrumb bar. The *component* references the page component that is displayed in the main area of the web interface.

Example:

```
version: "1.0"
type: route
data:
  url: "/services/clamav/onaccess-scans/create"
  title: _("Create")
  notificationTitle: _("Created on-access scan.")
  component: omv-services-clamav-onaccess-scan-form-page
```

### 9.5.3 Build configuration

To build and apply the final web interface configuration you need to run `omv-mkworkbench COMMAND` where `COMMAND` is `all | dashboard | log | navigation | route | i18n`.

CHAPTER

TEN

# SUPPORT

At present there are many ways of getting support, many of them are done by the community. The current support channels are:

**Code**

https://scm.openmediavault.org/

**Bugtracker**

http://bugtracker.openmediavault.org/

**Forum**

https://forum.openmediavault.org/

**IRC**

We have a support IRC channel at freenode servers, just type `/join #openmediavault` in your favourite IRC client, type your question and wait for someone available to help you.

**Facebook**

https://www.facebook.com/openmediavault/

**Twitter**

https://twitter.com/OpenMediaVault/

**Discord**

Since last year there is also a discord group. You can get access by clicking here.

**Note**

Make sure you provide as much information as you can when posting in the forum or bugtracker and describing your problem. If you have an error in the web interface make sure you take screenshots of the backtrace, to identify properly what's failing.

CHAPTER

ELEVEN

# PRESS RELEASES, REVIEWS AND EXTERNAL REFERENCES

- Dec 2008, FreeNAS: BSD Line and Linux Fork Linux magazine
- October 2011 "First version of the NAS distribution openmediavault" pro-linux.de (German)
- September 2014, "Community Choice" Project of the Month – openmediavault Sourceforge
- April 2015, Interview openmediavault developer Volker Theile Canox
- August 2015 LinuxVoice Magazine issue #9, Distrohopper LinuxVoice Magazine
- November 2014, openmediavault Open source network attached storage for Debian/GNU Linux ODROID Magazine
- January 2014 "How to build your own NAS box" APC Magazine Australia
- August 2015 "The open-source NAs distro for media lovers" `ACP Magazine Australia`
- Distribution Release: openmediavault 2.1 Distrowatch
- September 2014, openmediavault 1.0 review Linuxbsdos.com

CHAPTER

# TWELVE

# CONTRIBUTORS

**Founder and project manager**

Volker Theile

**Forum**

- area3o
- Cpoc
- davidh2k
- Dennis
- donh
- i814u2
- jensk
- jhmiller
- KM0201
- knumsi
- PhantomSens
- ryecoaaron
- SerErris
- Solo0815
- spyalelo
- subzeroin
- SVS
- tekkbebe
- WastlJ
- The Master

**Documentation**

- area3o
- Davidh2k
- GreenBean

- i814u2
- Reddy
- witopi
- subzero79

**Testing**

- Falk Menzel

**Translators**

- Alexandr aka azlk (Russian)
- Andrey Chapalda (Ukrainian)
- Antonio Pelaez Redondo (Spanish)
- Babchuk Volodymyr Romanovych (Ukrainian)
- Balajti Ádám (Hungarian)
- Bocquet Stéphane (French)
- Cyryl Sochacki (Polish)
- Gábor Majoros (Hungarian)
- Harry Stoker (Dutch)
- Helge Philipp (German)
- Henrik Sandström (Swedish)
- Jacek Niedziółka (Polish)
- Jakub Górny (Polish)
- Jonathan Weber (German)
- José Manuel Caínzos Sánchez (Spanish)
- Kochin Chang (Chinese (Taiwan))
- Kostas Gounaris (Greek)
- Marcel Beck (German)
- Mario Varelli (Italian)
- Mathias Grünewald (German)
- Mauro Rulli (Italian)
- Miguel Anxo Bouzada (Galician)
- Milan Toet (Dutch)
- Ming Song (Chinese)
- Nahir Mohamed (French)
- Nelson Rosado (Portuguese)
- Paolo Velati (Italian)
- Raul Fernandez Garcia (Spanish)
- Rune Bystrøm (Norwegian)

- Seba Kerckhof (Dutch)
- Serhat SUT (Turkish)
- Stefan Thrane Overby (Danish)
- Stephan Steiner (German)
- Szanyi Szabolcs (Hungarian)
- Tim Debie (Dutch)
- Tobias Brechle (German)
- Toshihiro Kan (Japanese)
- Volker Theile (German)
- Wei Ding (Chinese)
- (Russian)
- (Russian)
- (Russian)
- Maliar Benoit (French)
- Ji-Hyeon Gim (Korean)
- Sergio Rius Rodriguez (Spanish/Portuguese)
- Chang-Eon Byeon (Korean)
- Jan Štourač (Czech)
- Sungjin Kang (Korean)
- Pavel Borecki (Czech)
- Joaquim Farriol (Catalan)
- Plamen Vasilev (Bulgarian)
- Gábor Sári (Hungarian)
- Jonatan Nyberg (Swedish)
- Gyuris Gellért (Hungarian)
- Miha Bezjak (Slovenian)
- Jérémy D (French)
- Herald Yu (Chinese)
- Axel Cantenys (French)
- Abo Khalid (Arabic)
- Kevin Leclercq (French)
- Rudy ten Kate (Dutch)
- Jonatan Nyberg (Swedish)
- André Fernandes (Portuguese)
- Giovanni Scafora (Italian)
- Stephan Paternotte (Dutch)

**Code**

- Stefan Seidel
- Don Harpell
- Ralf Lindlein
- Tony Guepin
- Ian Grant

More code contributers can be found here.

CHAPTER

# THIRTEEN

# COPYRIGHT