FreedomBox Manual

Contents

1 FreedomBox: take your online privacy back

FreedomBox is a ready made personal server, designed with privacy and data ownership in mind. It is a subset of the Debian universal operating system and includes free software only. You can run it on a small, inexpensive and power-efficient computer box in your home that is dedicated for that use. It can also be installed on any computer running Debian or in a virtual machine.

In order to replace third-party communication services that are data mining your entire life, you will be able to host services yourself and use them at home or over the Internet through a browser or specialized apps. These services include chat and voice calls, webmail, file sharing and calendar, address book and news feed synchronization. For example, to start using a private chat service, activate the service from the administration interface and add your friends as authorized users of the service. They will be able to connect to the service hosted on your FreedomBox, using XMPP chat clients such as Conversations on Android, Pidgin on Windows and Linux, or Messages on Mac OS, for encrypted communications.

FreedomBox is a product you can just buy, set up and use. Once installed the interface is easy to use, similar to a smart phone.

User documentation:

- List of applications offered by FreedomBox.
- Manual
- Live Help from the community

FreedomBox can also host a Wi-Fi access point, ad blocking proxy and a virtual private network (VPN). More advanced users can replace their router with a FreedomBox.

Setting up FreedomBox on a specific hardware or on your computer running Debian may require a bit of technical expertise or help from the community.

Related technical documentation:

- Machines that support FreedomBox
- Download and Install
- FreedomBox Developer Manual

1.1 Typical usage: Private Cloud

FreedomBox provides services to the computers and mobile devices in your home, and to your friends. This includes secure instant messaging and low-bandwidth, high-quality voice conference calling. FreedomBox lets you publish your content in a blog and wiki to collaborate with the rest of the world. On the roadmap are a personal email server and federated social networking using GNU Social and Diaspora, to provide privacy-respecting alternatives to Gmail and Facebook.

1.2 Advanced usage: Smart Home Router

FreedomBox runs in a physical computer and can route your traffic. It can sit between various devices at home such as mobiles, laptops and TVs and the Internet, replacing a home wireless router. By routing traffic, FreedomBox can remove tracking advertisements and malicious web bugs before they ever reach your devices. FreedomBox can cloak your location and protect your anonymity by "onion routing" your traffic over Tor. FreedomBox provides a VPN server that you can use while you are away from home to keep your traffic secret on untrusted public wireless networks and to securely access various devices at home.

It can also be carried along with your laptop and used to connect to public networks at work, school or office to avail its services.

It could be used in a village to make available digital communications throughout the village. In the future, FreedomBox intends to deliver support for alternative ways of connecting to the Internet such as Mesh networking.

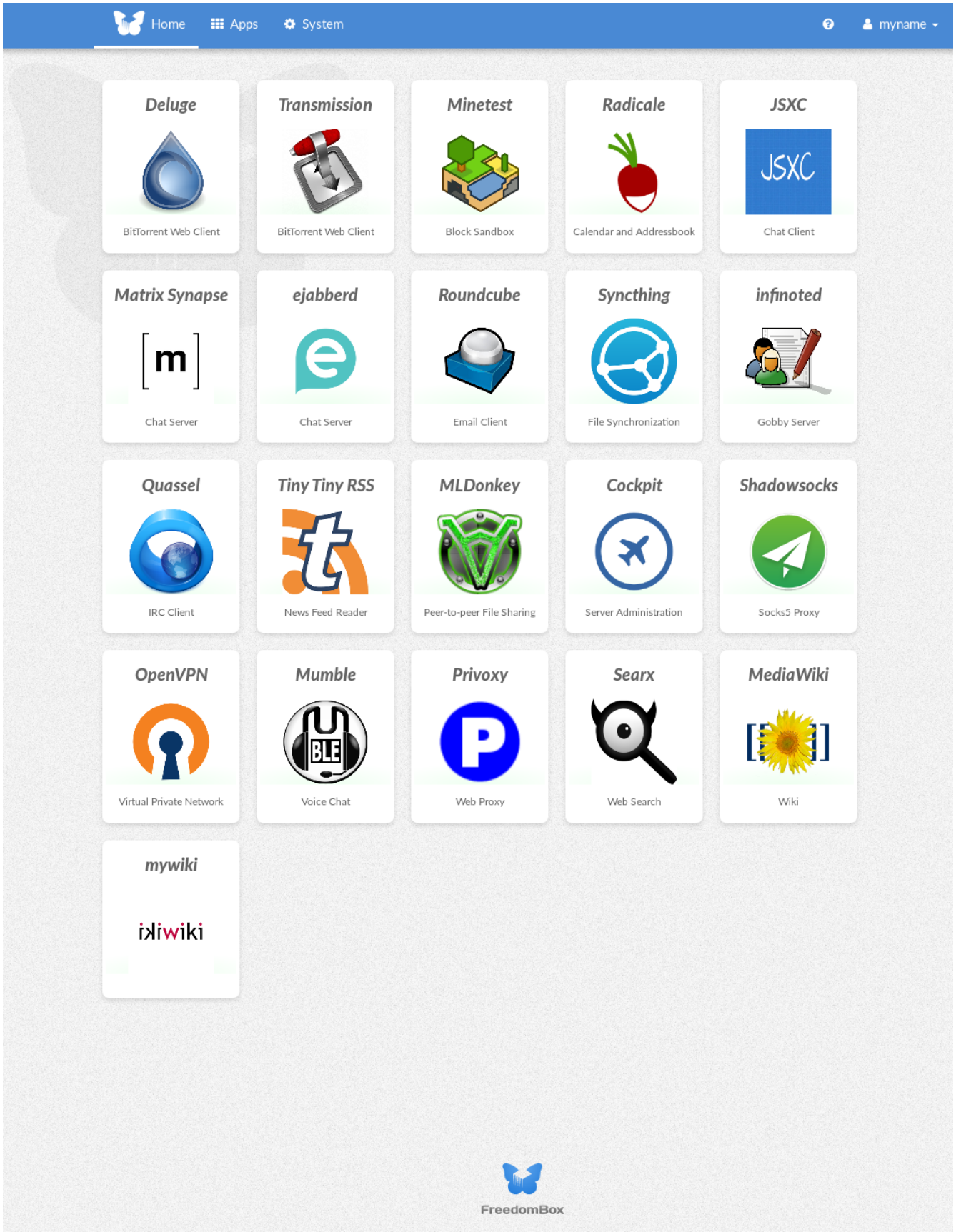
1.3 Advanced usage: For Communities

The primary design goal of FreedomBox is to be used as a personal server at home for use by a single family and their friends. However, at the core, it is a server software that can aid a non-technical user to setup services and maintain them with ease. Security is automatically managed and many of the technical choices in system administration are taken care by the software automatically thereby reducing complexity for a non-technical user. This nature of FreedomBox makes it well-suited for hosting services for small communities like villages or small firms. Communities can host their own services using FreedomBox with minimal effort. They can setup Wi-Fi networks that span the entire area of the community and draw Internet connections from long distances. Community members can enjoy previously unavailable Internet connectivity, ubiquitous Wi-Fi coverage, free VOIP services, offline education and entertainment content, etc. This will also boost privacy for individuals in the community, reduce dependence on centralized services provided by large companies and make them resistant to censorship.

The free e-book FreedomBox for Communities describes the motivation and provides detailed instructions to setup FreedomBox for this use case. Members of the FreedomBox project are involved in setting up Wi-Fi networks with free Internet connectivity in rural India. This e-book documents their knowledge and experiences.

1.4 FreedomBox Interface

1.4.1 Screenshot



1.4.2 Screencast introduction

Plinth_Introduction.webm

(36 MB, 13 Min.)

1.4.3 More video resources

Eben Moglen's talk, Eben Moglen - Freedom in the cloud, delivered before the FreedomBox project was started gives insights into the philosophy behind FreedomBox.

First demonstration of FreedomBox at SFLC, University of Columbia by Sunil Mohan Adapa.

2 Quick Start

2.1 What you need to get started

- A supported device (including any device that can run Debian). We will call that the FreedomBox in the rest of this manual.
- A power cable for your device.
- An ethernet cable.
- A microSD card (or equivalent storage media for your device), prepared according to the instructions on the Download page.

2.2 How to get started

1. Plug one end of your ethernet cord into your FreedomBox's ethernet port, and plug the other end into your router.
2. Power on the FreedomBox.
   - **Note:** On most single board computers, don't expect any output on a monitor connected via HDMI as the support may not exist in the kernel. See below to access and control your FreedomBox via network.
3. On first boot, FreedomBox will perform its initial setup (older versions of FreedomBox reboot after this step). This process may take several minutes on some machines. After giving it about 10 minutes, proceed to the next step.
   - **Note:** Currently, due a known bug, you need to restart your FreedomBox after 10m and then proceed to the next step.
4. After the FreedomBox has finished its initial setup, you can access its web interface (called Plinth) through your web browser.
   - If your computer is connected directly to the FreedomBox through a second (LAN) ethernet port, you can browse to: `http://freedombox/` or `http://10.42.0.1/`.
   - If your computer supports mDNS (GNU/Linux, Mac OSX or Windows with mDNS software installed), you can browse to: `http://freedombox.local/` (or `http://the-hostname-you-entered-during-install.local/`)
   - If you know your way around the router's web interface, you can look up the IP address of the FreedomBox there, and browse to that address.
   - If none of these methods are available, then you will need to figure out the IP address of your FreedomBox. You can use the "nmap" program from your computer to find its IP address:

     ```
     nmap -p 80 --open -sV 192.168.0.0/24 (replace the ip/netmask with the one the ↩
         router uses)
     ```

     In most cases you can look at your current IP address, and change the last digits with zero to find your home network, like so: XXX.XXX.XXX.0/24
     Your FreedomBox will show up as an IP address with an open tcp port 80 using Apache httpd service on Debian, such as the example below which would make it accessible at `http://192.168.0.165`:

```
Nmap scan report for 192.168.0.165
Host is up (0.00088s latency).
PORT   STATE SERVICE VERSION
80/tcp open  http    Apache httpd 2.4.17 ((Debian))
```

If nmap does not find anything with the above command, you can try replacing 192.168.0.0/24 with 10.42.0.255/24.

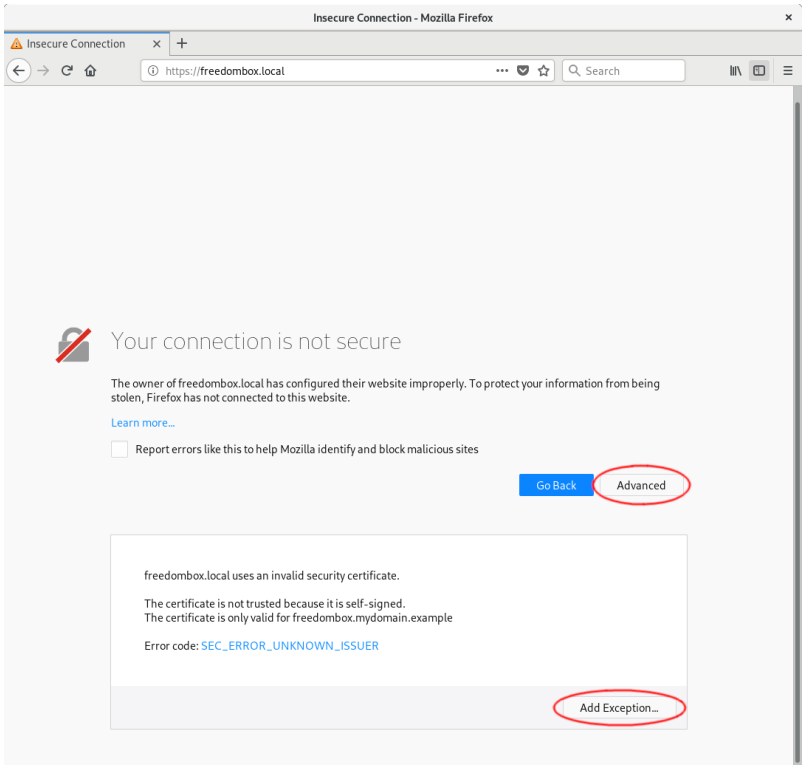
```
nmap -n -sP 10.42.0.255/24
```

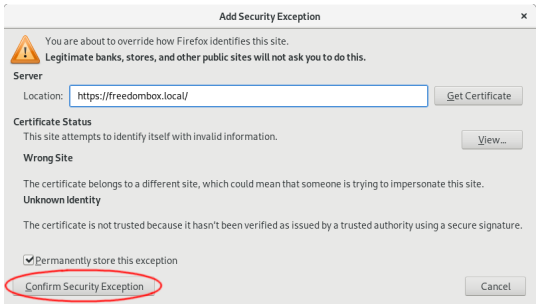
The scan report will show something similar to the following:

```
Nmap scan report for 10.42.0.1
Host is up (0.00027s latency).
Nmap scan report for 10.42.0.50
Host is up (0.00044s latency).
```

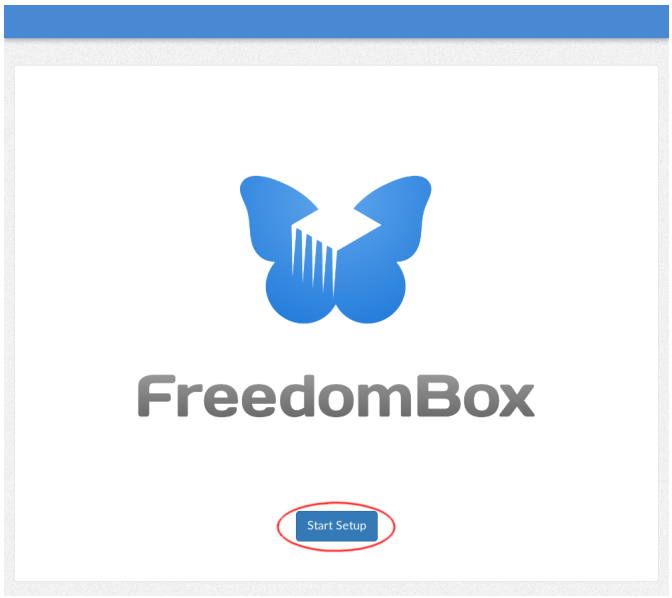
In this example, the FreedomBox is accessible at http://10.42.0.50. (10.42.0.1 is my laptop.)

5. On accessing FreedomBox's web interface (Plinth) your browser will warn you that it communicates securely but that it regards the security certificate for doing so as invalid. This is a fact you need to accept because the certificate is auto generated on the box and therefore "self-signed" (the browser might also use words such as "untrusted", "not private", "privacy error" or "unknown issuer/authority"). Telling your browser that you are aware of this might involve pressing buttons such as "I understand the Risks", "proceed to ... (unsafe)" or "Add exception". After installation this certificate can be changed to a normal one using the Let's Encrypt option.





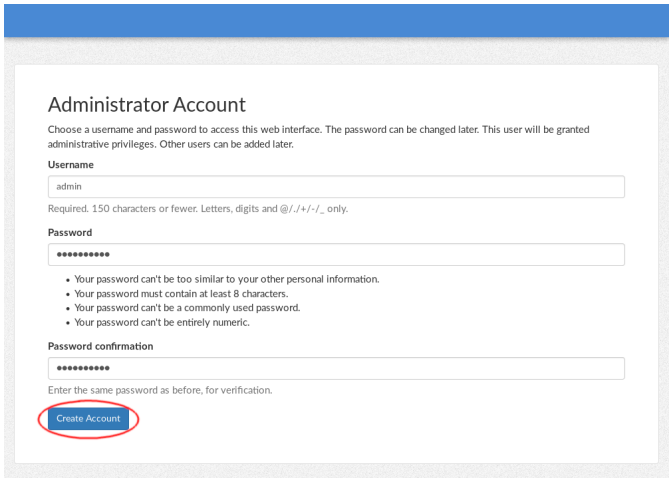
6. The first time you access the FreedomBox web interface, you will see a welcome page. Click the "Start Setup" button to continue.



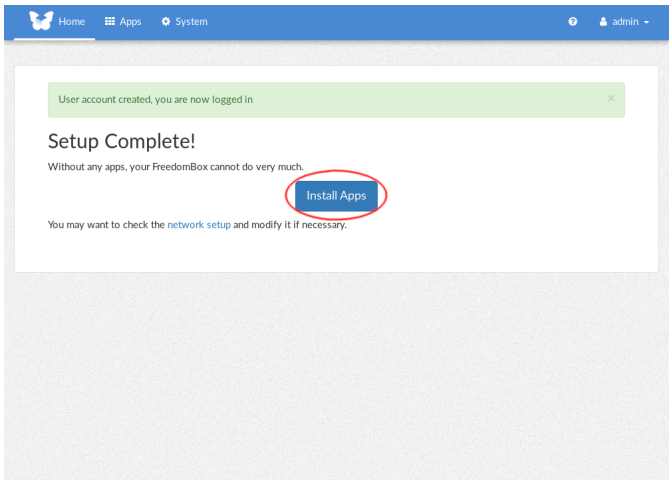
   - If you have installed FreedomBox using a Debian package, you will be asked for a secret key. This secret must have been provided during the installation of the Debian package. It can also be read from the file `/var/lib/plinth/firstboot-wi`

7. The next page asks you to provide a user name and password. Fill in the form, and then click "Create Account."

   - Note: The user that you create here has Admin privileges and can also log in using ssh. You might not want to use the user account you will want to use in daily usage, to prevent security issues. You can later add more users.



8. After completing the form, you will be logged in to FreedomBox's web interface (Plinth) and able to access apps and configuration through the interface.



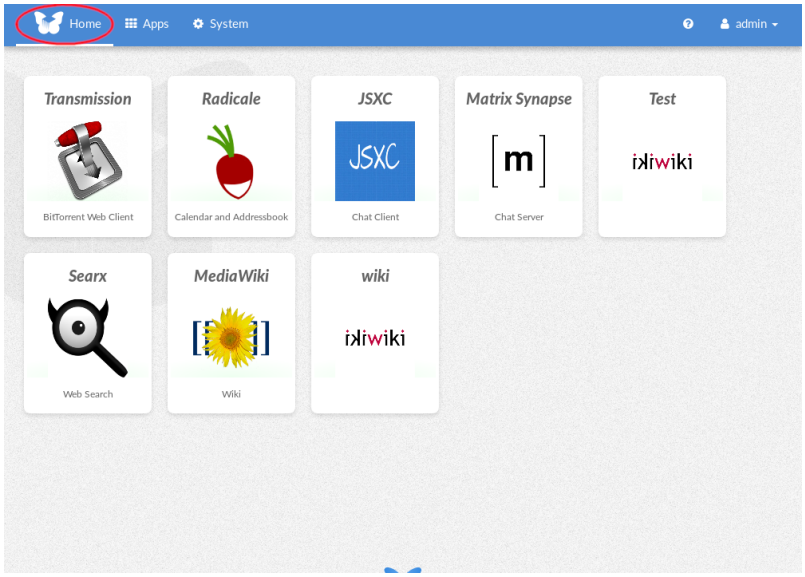
Now you can try any of the Apps that are available on FreedomBox.

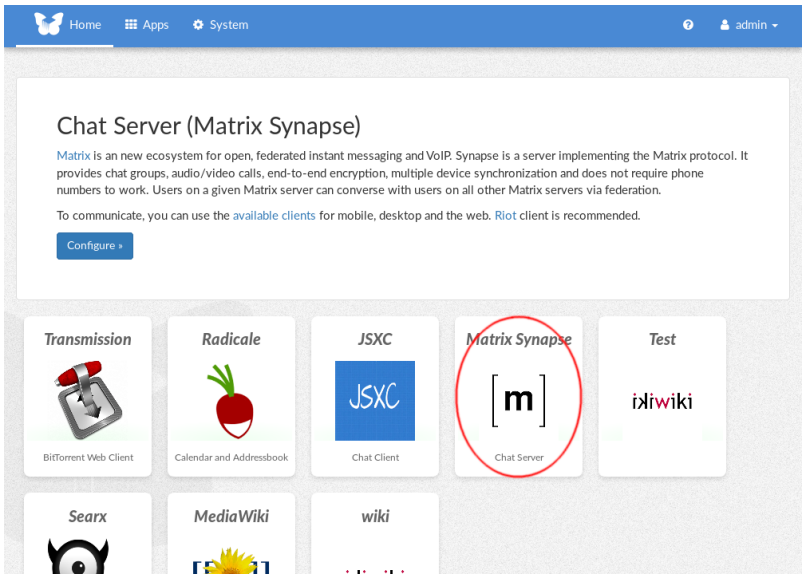
2.3 Finding your way around

2.3.1 Front page

The front page is the page that you will see when accessing the web root of your FreedomBox. You can also access it by clicking the FreedomBox logo in the top-left corner of the FreedomBox's web interface (Plinth).

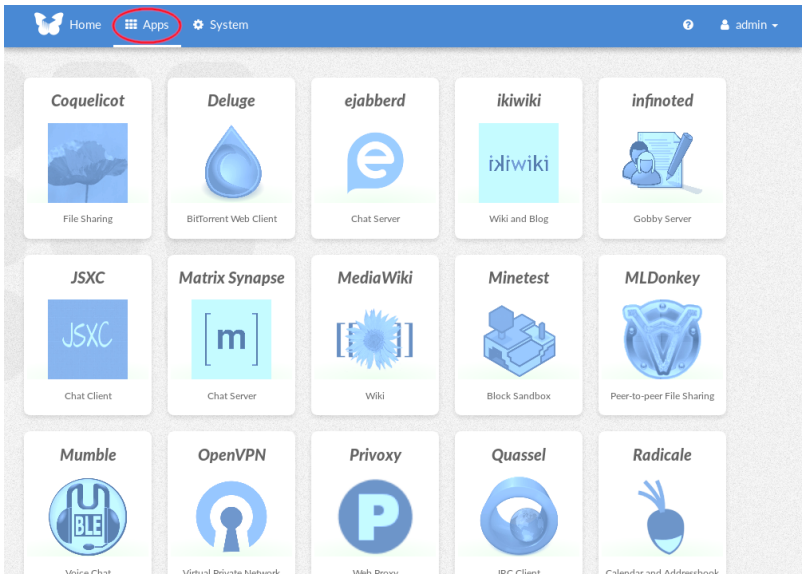
The front page includes shortcuts to apps that have been installed and are enabled. For web apps, clicking the shortcut will take you directly to the app's web page. For other services, clicking the shortcut will show more information about the service.





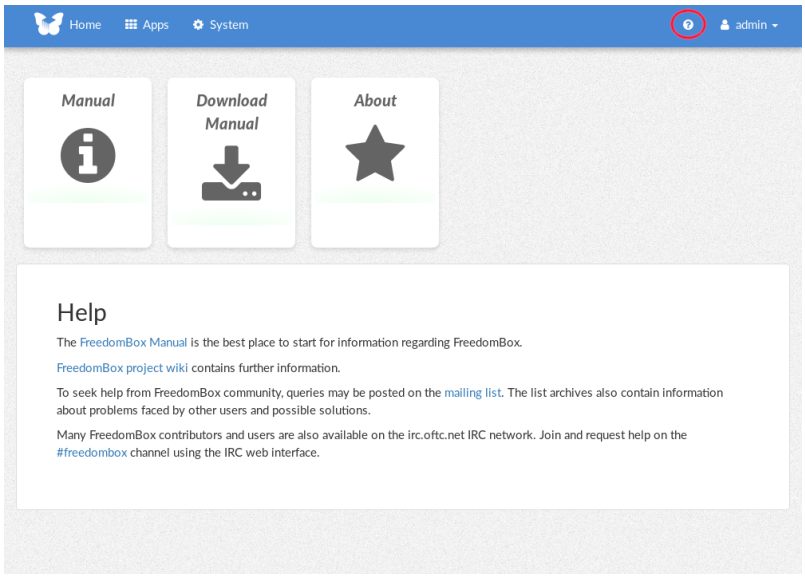
2.3.2 Apps menu

The Apps menu can be accessed by clicking the grid icon, next to the FreedomBox logo. This page lists all of the apps that are available for installing on FreedomBox. Click the name of an app to visit its page, where you can install and configure it.



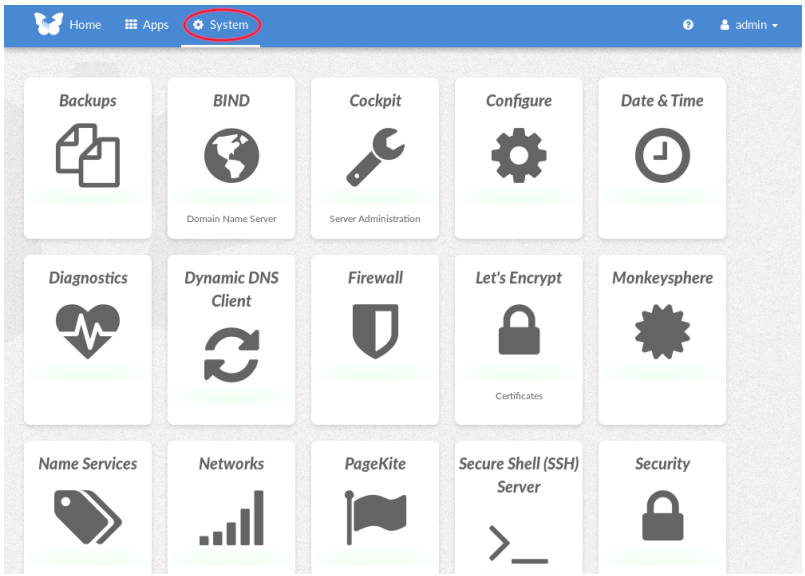
2.3.3 Help menu

The Help menu can be accessed by clicking the question mark icon in the top-right corner. It includes helpful links and the FreedomBox manual.



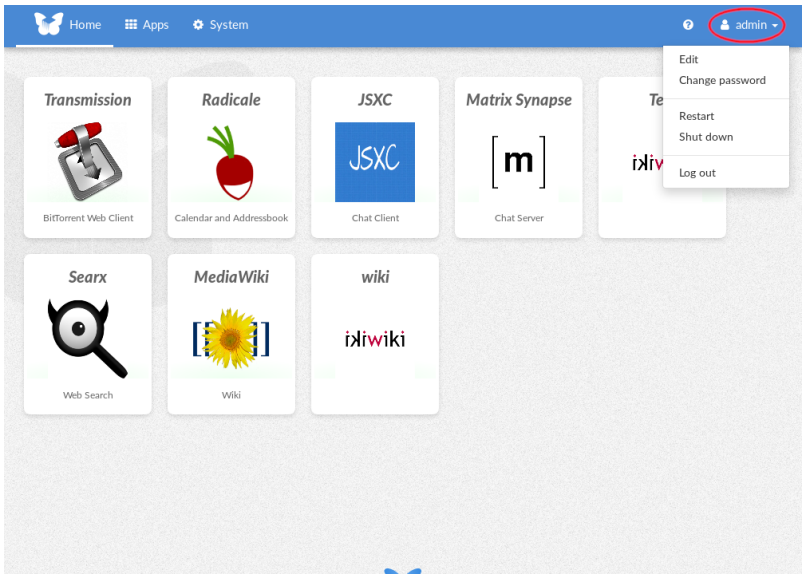
2.3.4 System menu

The System menu can be accessed by clicking the gear icon in the top-left corner. It includes a number of pages related to system configuration.



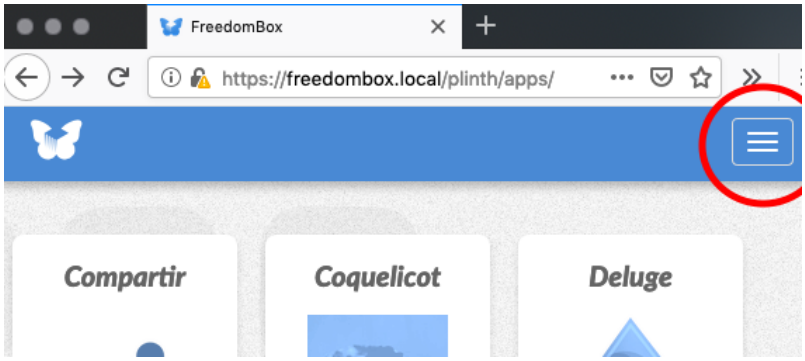
2.3.5 User menu

In the top-right corner, the name of the currently logged-in user is shown. A drop-down menu includes options for editing the current user or logging out of the user interface.



2.3.6 Burger menu

FreedomBox's web interface (Plinth) is responsive. Eventually you might miss the menu options on slim windows.



That's because the top menu options collapsed into the burger icon shown at the top right corner of the window. Clicking on it the menu drops down.

3 Getting Help

This manual is intended to give you the information you need to get started with your FreedomBox. However, if you have any questions after reading this document, you can get help from community contributors by:

- Searching/asking on our discussion forum (recommended).
- Emailing to our mailinglist at freedombox-discuss@lists.alioth.debian.org. You can also sign up to receive copies of every discussion that happens on the mailing list or read the archives.
- Chatting at #freedombox@irc.oftc.net.
- Reading the wiki.
- Reading the FreedomBox Foundation's website.
- Reading the FreedomBox Project Page.

4 Download and Install

Welcome to the FreedomBox download page. You may either install FreedomBox on one of the supported inexpensive hardware devices, on any Linux Debian operating system, or deploy it on a virtual machine.

Installing on a machine running a Debian system is easy because FreedomBox is available as a package. We do recommend to install FreedomBox on a supported single board computer (SBC). The board will be dedicated for FreedomBox use from home, this will prevent a lot of risks, such as accidental misconfiguration by the user. In case of trouble deciding which hardware is best for you or during the installation, please use the support page or read the Questions and Answers page based on posts on the Freedombox-discuss mailing list archives.

4.1 Downloading on Debian

If you are installing on an existing Debian installation, you don't need to download these images. Instead read the instructions on setting up FreedomBox on Debian.

4.2 Downloading for SBC or Virtual Machine

4.2.1 Prepare your device

Read the hardware specific instructions on how to prepare your device at the Hardware section. On the web is a lot of documentation about setting your device up and flashing USB or SD Cards to boot your hardware.

4.2.2 Downloading Images

Recent images for supported targets are available here:

- Official Images: `https://freedombox.org/download/`
- Official Images: `https://ftp.freedombox.org/pub/freedombox/`

4.2.3 Verifying the Downloaded Images

It is important to verify the images you have downloaded to ensure that the file has not be corrupted during the transmission and that it is indeed the image built by FreedomBox developers.

Note: Testing and nightly images are automatically signed by the FreedomBox CI server.

- First open a terminal and import the public keys of the FreedomBox developers who built the images:

```
$ gpg --recv-keys BCBEBD57A11F70B23782BC5736C361440C9BC971

$ gpg --recv-keys 7D6ADB750F91085589484BE677C0C75E7B650808

# This is the FreedomBox CI server's key
$ gpg --recv-keys 013D86D8BA32EAB4A6691BF85D4153D6FE188FC8
```

  If this command shows an error such as *new key but contains no user ID - skipped*, then use a different keyserver to download the keys:

```
$ gpg --keyserver keys.gnupg.net --recv-keys BCBEBD57A11F70B23782BC5736C361440C9BC971
$ gpg --keyserver keys.gnupg.net --recv-keys 7D6ADB750F91085589484BE677C0C75E7B650808
$ gpg --keyserver keys.gnupg.net --recv-keys 013D86D8BA32EAB4A6691BF85D4153D6FE188FC8
```

  Or

```
$ gpg --keyserver keyserver.ubuntu.com --recv-keys  ↩
    BCBEBD57A11F70B23782BC5736C361440C9BC971
$ gpg --keyserver keyserver.ubuntu.com --recv-keys 7 ↩
    D6ADB750F91085589484BE677C0C75E7B650808
$ gpg --keyserver keyserver.ubuntu.com --recv-keys 013 ↩
    D86D8BA32EAB4A6691BF85D4153D6FE188FC8
```

- Next, verify the fingerprint of the public keys:

```
$ gpg --fingerprint BCBEBD57A11F70B23782BC5736C361440C9BC971
pub   4096R/0C9BC971 2011-11-12
      Key fingerprint = BCBE BD57 A11F 70B2 3782  BC57 36C3 6144 0C9B C971
uid                  Sunil Mohan Adapa <sunil@medhas.org>
sub   4096R/4C1D4B57 2011-11-12

$ gpg --fingerprint 7D6ADB750F91085589484BE677C0C75E7B650808
pub   4096R/7B650808 2015-06-07 [expires: 2020-06-05]
      Key fingerprint = 7D6A DB75 0F91 0855 8948  4BE6 77C0 C75E 7B65 0808
uid                  James Valleroy <jvalleroy@mailbox.org>
uid                  James Valleroy <jvalleroy@freedombox.org>
sub   4096R/25D22BF4 2015-06-07 [expires: 2020-06-05]
sub   4096R/DDA11207 2015-07-03 [expires: 2020-07-01]
sub   2048R/2A624357 2015-12-22

$ gpg --fingerprint 013D86D8BA32EAB4A6691BF85D4153D6FE188FC8
pub   rsa4096 2018-06-06 [SC]
      013D 86D8 BA32 EAB4 A669  1BF8 5D41 53D6 FE18 8FC8
uid           [ unknown] FreedomBox CI (Continuous Integration server) <admin@freedombox. ↩
    org>
sub   rsa4096 2018-06-06 [E]
```

- Finally, verify your downloaded image with its signature file `.sig`. For example:

```
$ gpg --verify freedombox-unstable-free_2015-12-13_cubietruck-armhf.img.xz.sig freedombox- ↩
    unstable-free_2015-12-13_cubietruck-armhf.img.xz
gpg: Signature made Thursday 15 January 2015 09:27:50 AM IST using RSA key ID 0C9BC971
gpg: Good signature from "Sunil Mohan Adapa <sunil@medhas.org>"
```

```
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: BCBE BD57 A11F 70B2 3782  BC57 36C3 6144 0C9B C971
```

4.2.4 Installation

After the download you can use the image to boot your chosen hardware (including virtual machines). You'll need to copy the image to the memory card or USB stick as follows:

1. Figure out which device your card actually is.
    1. Unplug your card.
    2. Run `dmesg -w` to show and follow the kernel messages.
    3. Plug your card in. You will see messages such as following:

```
[33299.023096] usb 4-6: new high-speed USB device number 12 using ehci-pci
[33299.157160] usb 4-6: New USB device found, idVendor=058f, idProduct=6361
[33299.157162] usb 4-6: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[33299.157164] usb 4-6: Product: Mass Storage Device
[33299.157165] usb 4-6: Manufacturer: Generic
[33299.157167] usb 4-6: SerialNumber: XXXXXXXXXXXX
[33299.157452] usb-storage 4-6:1.0: USB Mass Storage device detected
[33299.157683] scsi host13: usb-storage 4-6:1.0
[33300.155626] scsi 13:0:0:0: Direct-Access     Generic- Compact Flash    1.01 PQ:  ↩
    0 ANSI: 0
[33300.156223] scsi 13:0:0:1: Direct-Access     Multiple Flash Reader     1.05 PQ:  ↩
    0 ANSI: 0
[33300.157059] sd 13:0:0:0: Attached scsi generic sg4 type 0
[33300.157462] sd 13:0:0:1: Attached scsi generic sg5 type 0
[33300.462115] sd 13:0:0:1: [sdg] 30367744 512-byte logical blocks: (15.5 GB/14.4  ↩
    GiB)
[33300.464144] sd 13:0:0:1: [sdg] Write Protect is off
[33300.464159] sd 13:0:0:1: [sdg] Mode Sense: 03 00 00 00
[33300.465896] sd 13:0:0:1: [sdg] No Caching mode page found
[33300.465912] sd 13:0:0:1: [sdg] Assuming drive cache: write through
[33300.470489] sd 13:0:0:0: [sdf] Attached SCSI removable disk
[33300.479493]  sdg: sdg1
[33300.483566] sd 13:0:0:1: [sdg] Attached SCSI removable disk
```

    4. In the above case, the disk that is newly inserted is available as */dev/sdg*. Very carefully note this and use it in the copying step below.

2. Decompress the downloaded image using tar:

```
$ xz -d freedombox-unstable-free_2015-12-13_cubietruck-armhf.img.xz
```

   The above command is an example for the *cubietruck* image built on 2015-12-13. Your downloaded file name will be different.

3. Copy the image to your card. Double check to make sure you don't write to your computer's main storage (such as /dev/sda). Also make sure that you don't run this step as root to avoid potentially overriding data on your hard drive due to a mistake in identifying the device or errors while typing the command. USB disks and SD cards inserted into the system should typically be write accessible to normal users. If you don't have permission to write to your SD card as a user, you may need to run this command as root. In this case triple check everything before you run the command. Another safety precaution is to unplug all external disks except the SD card before running the command.

   For example, if your SD card is */dev/sdg* as noted in the first step above, then to copy the image, run:

```
$ dd bs=1M if=freedombox-unstable-free_2015-12-13_cubietruck-armhf.img of=/dev/sdg conv ↩
    =fdatasync status=progress
```

An alternative to copy to SD card command

- ```
  $ cat freedombox-unstable-free_2015-12-13_cubietruck-armhf.img > /dev/sdg ; sync
  ```

On MS Windows you will need a tool like *etcher*. On MacOS (OSX) you can use programs like *balenaetcher* and *rosaimagewriter*.

- The above command is an example for the *cubietruck* image built on 2015-12-13. Your image file name will be different.

  When picking a device, use the drive-letter destination, like */dev/sdg*, not a numbered destination, like */dev/sdg1*. The device without a number refers to the entire device, while the device with a number refers to a specific partition. We want to use the whole device. Downloaded images contain complete information about how many partitions there should be, their sizes and types. You don't have to format your SD card or create partitions. All the data on the SD card will be wiped off during the write process.

- Use the image by inserting the SD card or USB disk into the target device and booting from it. Your device should also be prepared (see the Hardware section).

- Read (the rest of) the Manual for instructions on how to use applications in FreedomBox.

4.3 Obtaining Source Code

FreedomBox is fully free software and you can obtain the source code to study, modify and distribute improvements.

4.3.1 From within FreedomBox

FreedomBox is made up of several software programs and you can obtain the source code to any of them. These instructions are similar to obtaining and building source code for Debian since FreedomBox is a pure blend of Debian. Using this process you can obtain the source code to the exact version of the package you are currently using in FreedomBox.

1. To see the list of software packages installed on your FreedomBox, run the following in a terminal:

   ```
   dpkg -l
   ```

2. To obtain the source code for any of those programs, then run:

   ```
   apt source <package_name>
   ```

   This requires that the apt sources list contains information about the source code repositories. These are present by default on all FreedomBox images. If you have installed FreedomBox using a package from Debian, you need to ensure that source repositories are added in the file.

3. To build the package from source code, first install its dependencies

   ```
   apt build-dep <package_name>
   ```

   Switch to the source directory created by the *apt source* command:

   ```
   cd <source_directory>
   ```

   Then build the package

   ```
   dpkg-buildpackage -rfakeroot -uc
   ```

4. Install the package:

   ```
   dpkg -i ../<built_package>.deb
   ```
   ```

4.3.2 Other Ways to Obtain Source Code

1. Source code for any of the packages can be browsed and searched using the web interface at sources.debian.org. For example, see the plinth package.
2. Source code and pre-built binary package for any version of a package including historic versions can be obtained from snapshot.debian.org. For example, see the plinth package.
3. You can also obtain the links to upstream project homepage, upstream version control, Debian's version control, changelog, etc. from the Debian tracker page for a project at tracker.debian.org. For example, see the tracker page for plinth package.
4. You can build and install a package from its Debian's version control repository. For example,

```
git clone https://salsa.debian.org/freedombox-team/plinth
cd plinth
apt build-dep .
dpkg-buildpackage -rfakeroot -uc
dpkg -i ../plinth*.deb
```

4.3.3 Building Disk Images

You can also build FreedomBox disk images for various hardware platforms using the freedom-maker tool. This is also available as a Debian package and source code for it may be obtained using the above methods. Build instructions for creating disk images are available as part of the source code for freedom-maker package.

FreedomBox disk images are built and uploaded to official servers using automated Continuous Integration infrastructure. This infrastructure is available as source code too and provides accurate information on how FreedomBox images are built.

4.3.3.1 U-boot on Pioneer Edition Images

There is one minor exception to the u-boot package present on the hardware sold as FreedomBox Home Server Kits Pioneer Edition. It contains an small but important fix that is not part of Debian sources. The fork of the Debian u-boot source repository along with the minor change done by the FreedomBox is available as a separate repository. We except this change to be available in upstream u-boot eventually and this repository will not be needed. This package can be built on a Debian armhf machine as follows (cross compiling is also possible, simply follow instructions for cross compiling Debian packages):

```
apt install git git-buildpackage
git clone https://salsa.debian.org/freedombox-team/u-boot.git
cd u-boot
pbuilder create --distribution=buster
gbp buildpackage --git-pbuilder
```

The u-boot Debian package will be available in *u-boot-sunxi*.deb*. This package will contain

```
mkdir temp
dpkg -x u-boot-suxi*.deb temp
unxz <lime2_image_built_with_freedom_maker>
dd if=temp/usr/lib/u-boot/A20-OLinuXino-Lime2/u-boot-sunxi-with-spl.bin of=<lime2.img> seek ↩
    =8 bs=1k conv=notrunc
```

The resulting image will have the modified u-boot in it.

5 Apps

5.1 User websites (userdir)

5.1.1 What is User websites?

User websites is a module of the Apache webserver enabled to allow users defined in the FreedomBox system to expose a set of static files on the FreedomBox filesystem as a website to the local network and/or the internet according to the network and

firewall setup.

Application basics	
Category	File sharing
Available since version	0.9.4
Upstream project website	https://httpd.apache.org/docs/2.4/mod/mod_userdir.html
Upstream end user documentation	https://httpd.apache.org/docs/2.4/howto/public_html.html

Screenshot

5.1.3 Using User websites

The module is always enabled and offers no configuration from the FreedomBox web interface. There is no configuration or status page shown for this module in the FreedomBox web interface.

To serve documents, place the files in the designated directory in a FreedomBox user's home directory in the filesystem.

This directory is: **public_html**

Thus the absolute path for the directory of a user named fbx with home directory in /home/fbx will be **/home/fbx/public_html**. User websites will serve documents placed in this directory when requests for documents with the URI path "~fbx" are received. For the the example.org domain thus a request for the document example.org/~fbx/index.html will transfer the file in /home/fbx/public_html/index.html.

Using SFTP to create public_html and upload documents

5.2 Anonymity Network (Tor)

5.2.1 What is Tor?

Tor is a network of servers operated by volunteers. It allows users of these servers to improve their privacy and security while surfing on the Internet. You and your friends are able to access to your FreedomBox via Tor network without revealing its IP address. Activating Tor application on your FreedomBox, you will be able to offer remote services (chat, wiki, file sharing, etc...) without showing your location. This application will give you a better protection than a public web server because you will be less exposed to intrusive people on the web.

5.2.2 Using Tor to browse anonymously

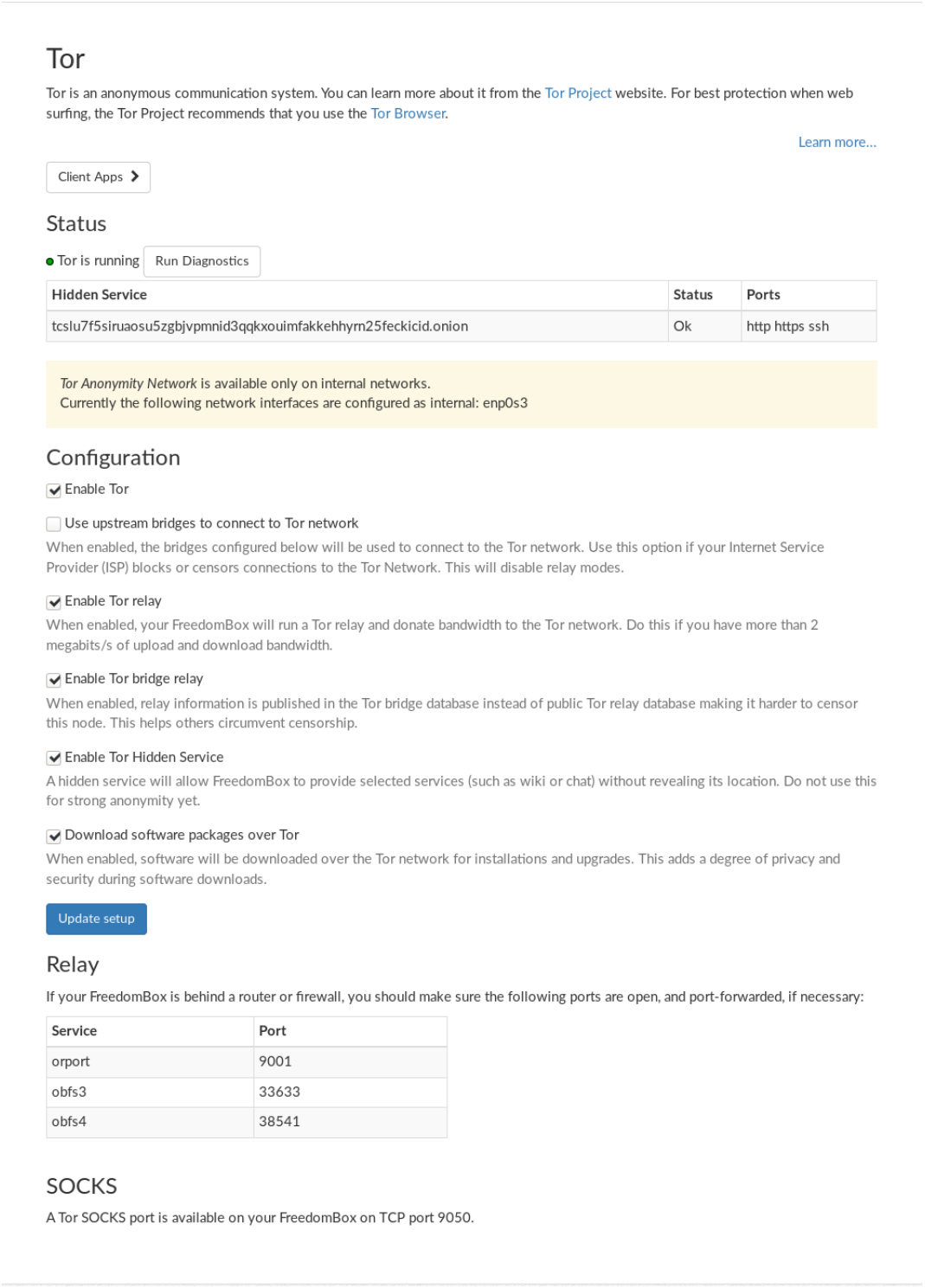
Tor Browser is the recommended way to browse the web using Tor. You can download the Tor Browser from https://www.torproject.org/projects/torbrowser.html and follow the instructions on that site to install and run it.

5.2.3 Using Tor Onion Service to access your FreedomBox

Tor Onion Service provides a way to access your FreedomBox, even if it's behind a router, firewall, or carrier-grade NAT (i.e., your Internet Service Provider does not provide a public IPv4 address for your router).

To enable Tor Onion Service, first navigate to the Anonymity Network (Tor) page. (If you don't see it, click on the FreedomBox logo at the top-left of the page, to go to the main Apps page.) On the Anonymity Network (Tor) page, under Configuration, check "Enable Tor Onion Service", then press the Update setup button. Tor will be reconfigured and restarted.

After a while, the page will refresh and under Status, you will see a table listing the Onion Service .onion address. Copy the entire address (ending in .onion) and paste it into the Tor Browser's address field, and you should be able to access your FreedomBox. (You may see a certificate warning because FreedomBox has a self-signed certificate.)



Currently only HTTP (port 80), HTTPS (port 443), and SSH (port 22) are accessible through the Tor Onion Service configured on the FreedomBox.

5.2.4 Apps accessible via Tor

The following apps can be accessed over Tor. Note that this list is not exhaustive.

- Calendar and Addressbook (Radicale)
- File Synchronization (Syncthing)

- Feed reader (TinyTinyRSS)
- Web Search (Searx)
- Wiki (MediaWiki)
- Wiki and Blog (Ikiwiki)

5.2.5 Running a Tor relay

When Tor is installed, it is configured by default to run as a bridge relay. The relay or bridge option can be disabled through the Tor configuration page in Plinth.

At the bottom of the Tor page in Plinth, there is a list of ports used by the Tor relay. If your FreedomBox is behind a router, you will need to configure port forwarding on your router so that these ports can be reached from the public Internet.

The requirements to run a relay are listed in the Tor Relay Guide. In short, it is

- recommended that a relay has at least 16 Mbit/s (Mbps) upload and download bandwidth available for Tor. More is better.
- required that a Tor relay be allowed to use a minimum of 100 GByte of outbound and of incoming traffic per month.
- recommended that a <40 Mbit/s non-exit relay should have at least 512 MB of RAM available; A relay faster than 40 Mbit/s should have at least 1 GB of RAM.

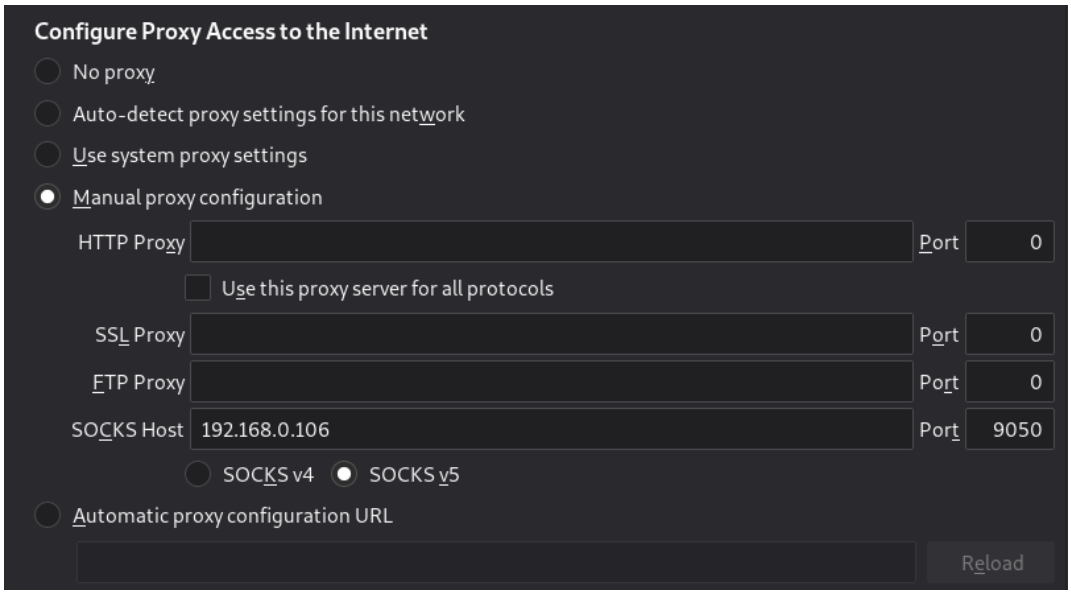
5.2.6 (Advanced) Usage as a SOCKS proxy

FreedomBox provides a Tor SOCKS port that other applications can connect to, in order to route their traffic over the Tor network. This port is accessible on any interfaces configured in the internal firewall zone. To configure the application, set SOCKS Host to the internal network connection's IP address, and set the SOCKS Port to 9050.

5.2.6.1 Example with Firefox

Your web browser can be configured to use the Tor network for all of your browsing activity. This allows for censorship circumvention and also hides your IP address from websites during regular browsing. For anonymity, using tor browser is recommended.

Configure your local FreedomBox IP address and port 9050 as a SOCKS v5 proxy in Firefox. There are extensions to allow for easily turning the proxy on and off.

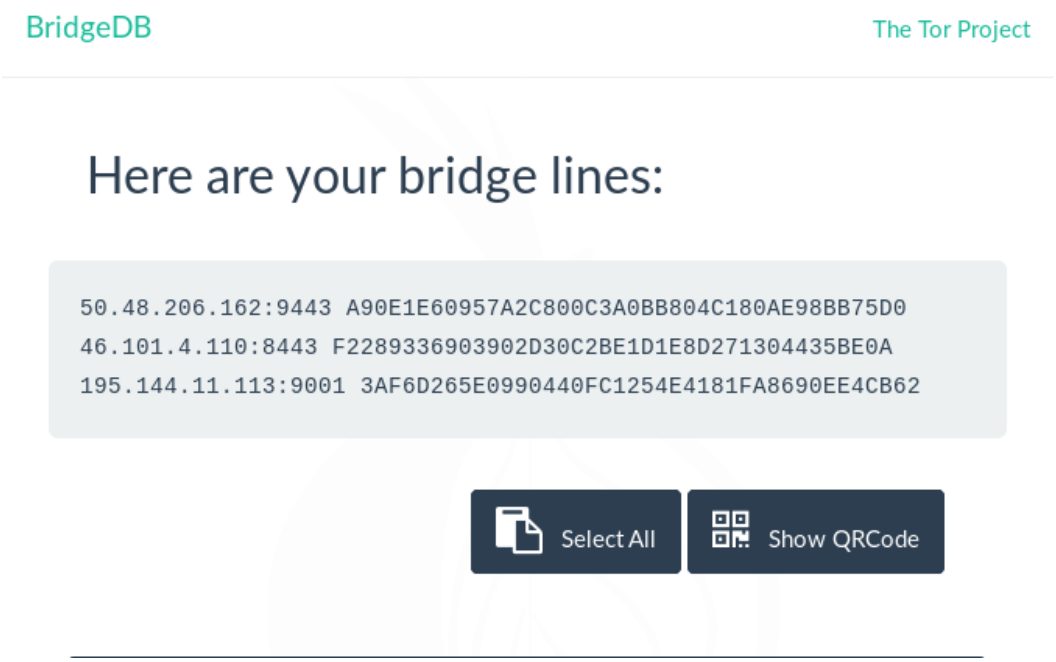


With the SOCKS proxy configured, you can now access any onion URL directly from Firefox. FreedomBox itself has an onion v3 address that you can connect to over the Tor network (bookmark this for use in emergency situations).

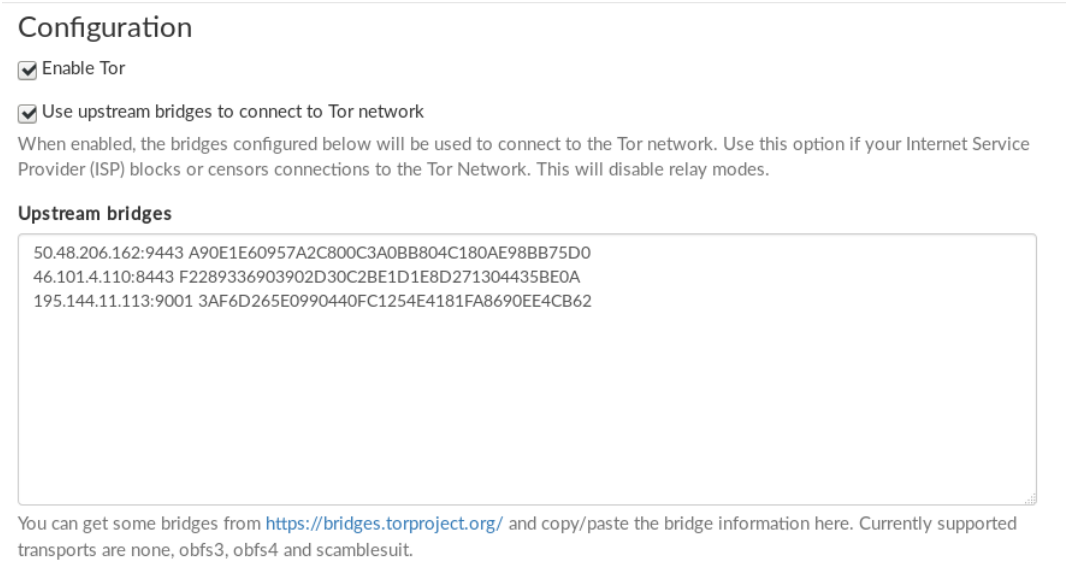
5.2.7 Circumventing Tor censorship

If your ISP is trying to block Tor traffic, you can use tor bridge relays to connect to the tor network.

1. Get the bridge configuration from the Tor BridgeDB



2. Add the lines to your FreedomBox Tor configuration as show below.



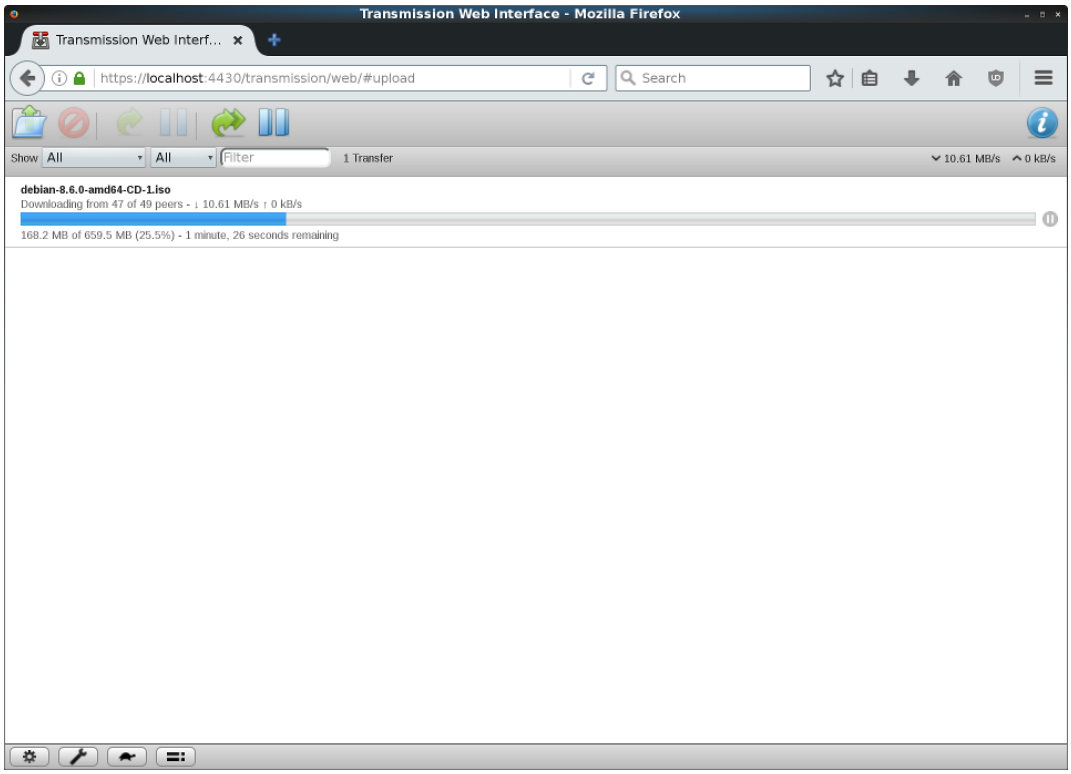
5.3 BitTorrent (Transmission)

5.3.1 What is Transmission ?

BitTorrent is a communications protocol using peer-to-peer (P2P) file sharing. It is not anonymous; you should assume that others can see what files you are sharing. There are two BitTorrent web clients available in FreedomBox: Transmission and Deluge. They have similar features, but you may prefer one over the other.

Transmission is a lightweight BitTorrent client that is well known for its simplicity and a default configuration that "Just Works".

5.3.2 Screenshot



5.3.3 Using Transmission

After installing Transmission, it can be accessed at `https://<your freedombox>/transmission`. Transmission uses single sign-on from FreedomBox, which means that if you are logged in on your FreedomBox, you can directly access Transmission without having to enter the credentials again. Otherwise, you will be prompted to login first and then redirected to the Transmission app.

5.3.4 Tips

5.3.4.1 Transferring Downloads from the FreedomBox

1. Transmission's downloads directory can be added as a shared folder in the "Sharing" app. You can then access your downloads from this shared folder using a web browser.

2. (Advanced) If you have the ssh access to your FreedomBox, you can use sftp to browse the downloads directory using a suitable file manager or web browser (e.g. dolphin or Konqueror).

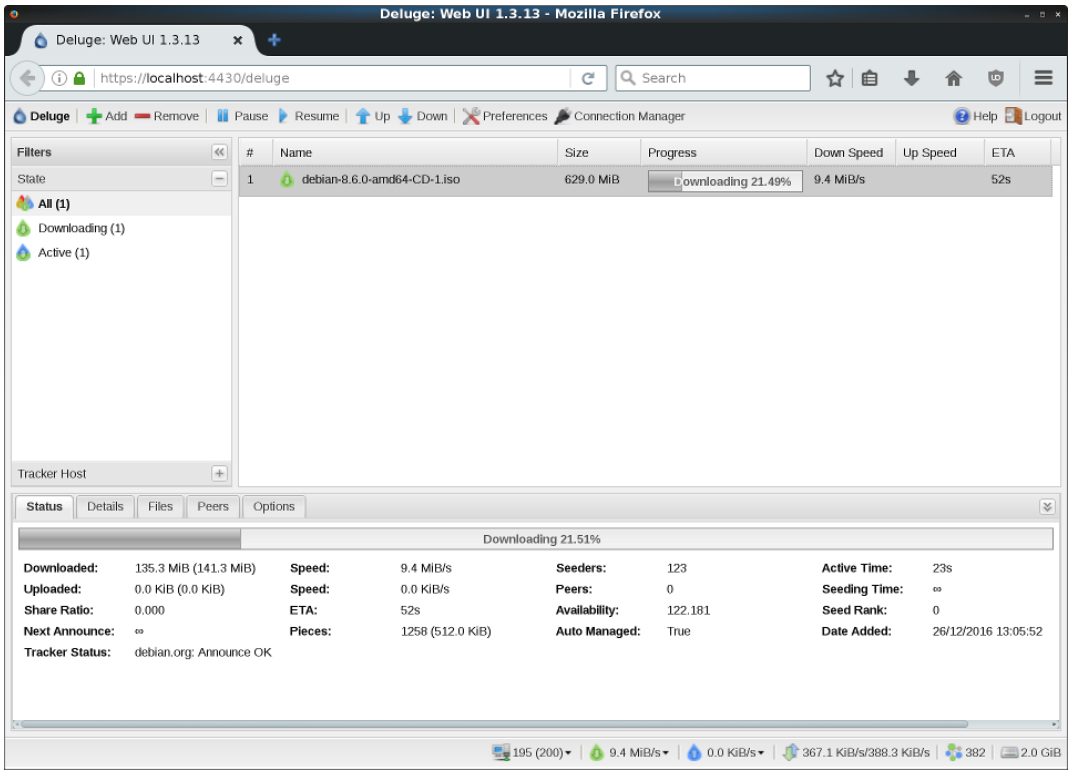
5.4 BitTorrent (Deluge)

5.4.1 What is Deluge?

BitTorrent is a communications protocol using peer-to-peer (P2P) file sharing. It is not anonymous; you should assume that others can see what files you are sharing. There are two BitTorrent web clients available in FreedomBox: Transmission and Deluge. They have similar features, but you may prefer one over the other.

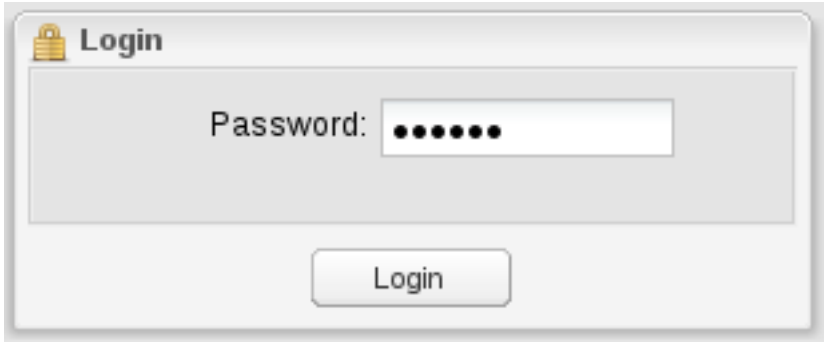
Deluge is a lightweight BitTorrent client that is highly configurable. Additional functionality can be added by installing plugins.

5.4.2 Screenshot



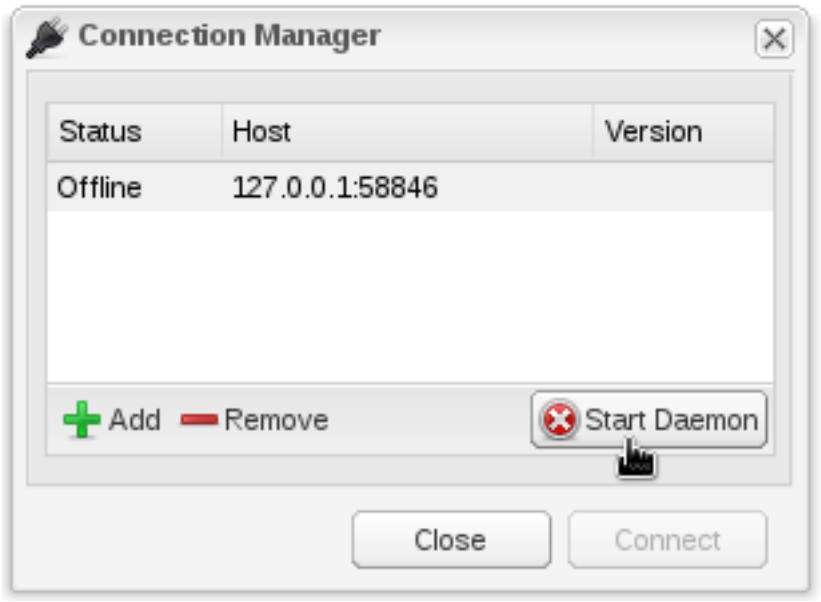
5.4.3 Initial Setup

After installing Deluge, it can be accessed by pointing your browser to `https://<your freedombox>/deluge`. You will need to enter a password to login:

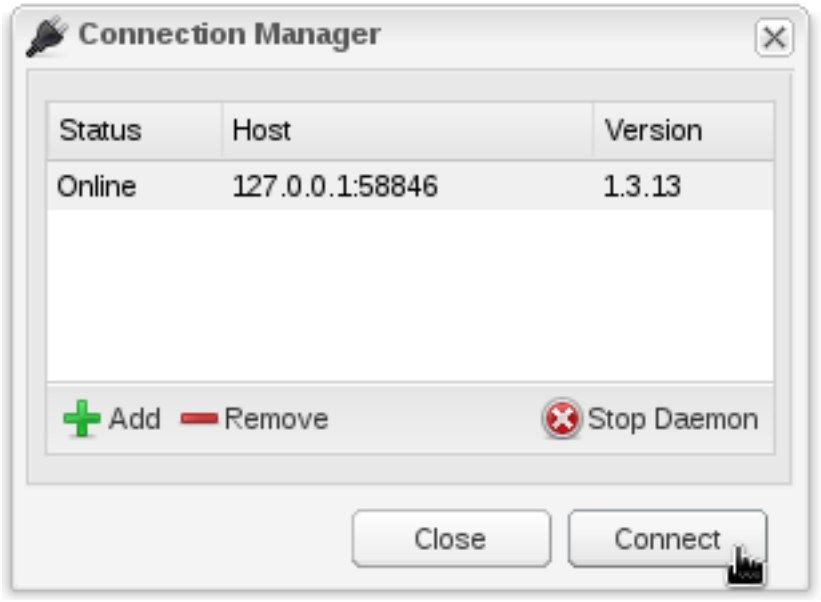


The initial password is "deluge". The first time that you login, Deluge will ask if you wish to change the password. You should change it to something that is harder to guess.

Next you will be shown the connection manager. Click on the first entry (Offline - 127.0.0.1:58846). Then click "Start Daemon" to start the Deluge service that will run in the background.



Now it should say "Online". Click "Connect" to complete the setup.



At this point, you are ready to begin using Deluge. You can make further changes in the Preferences, or add a torrent file or URL.

5.5 Block Sandbox (Minetest)

Minetest is a multiplayer infinite-world block sandbox. This module enables the Minetest server to be run on this FreedomBox, on the default port (30000). To connect to the server, a Minetest client is needed.

5.5.1 Port Forwarding

If your FreedomBox is behind a router, you will need to set up port forwarding on your router. You should forward the following ports for Minetest:

- UDP 30000

5.6 Calendar and Addressbook (Radicale)

With Radicale, you can synchronize your personal calendars, ToDo lists, and addressbooks with your various computers, tablets, and smartphones, and share them with friends, without letting third parties know your personal schedule or contacts.

5.6.1 Why should I run Radicale?

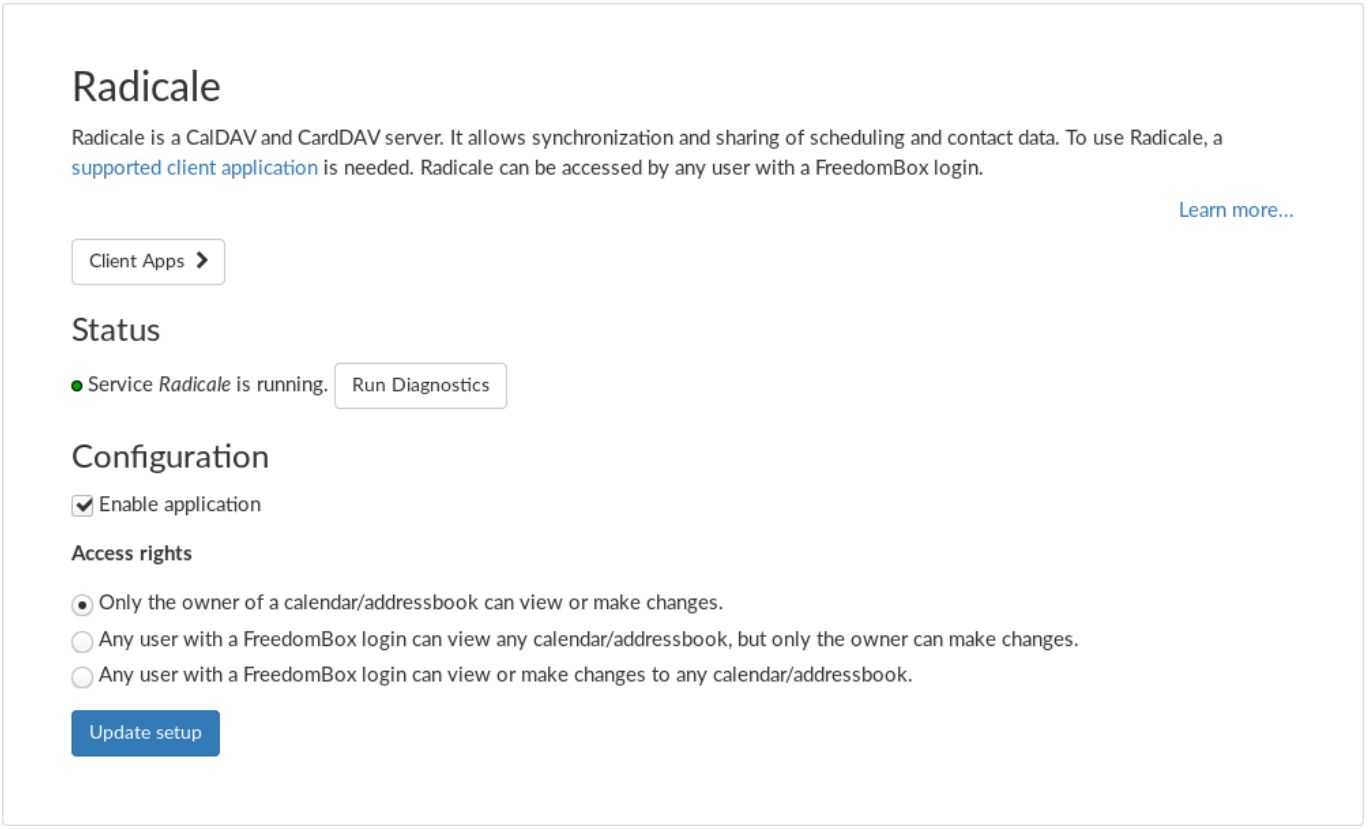
Using Radicale, you can get rid of centralized services like Google Calendar or Apple Calendar (iCloud) data mining your events and social connections.

5.6.2 How to setup Radicale?

First, the Radicale server needs to be activated on your box.

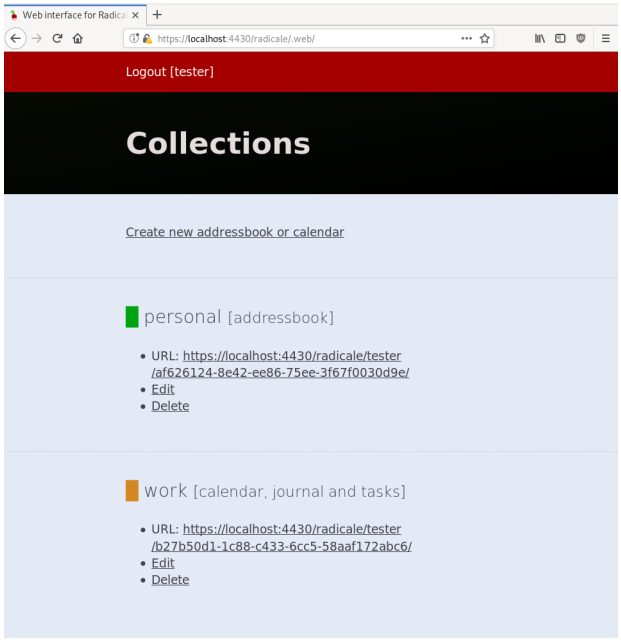
- Within FreedomBox Service (Plinth)
  1. select *Apps*
  2. go to *Radicale (Calendar and Addressbook)* and
  3. install the application. After the installation is complete, make sure the application is marked "enabled" in the FreedomBox interface. Enabling the application launches the Radicale CalDAV/CardDAV server.
  4. define the access rights:
     - Only the owner of a calendar/addressbook can view or make changes
     - Any user can view any calendar/addressbook, but only the owner can make changes
     - Any user can view or make changes to any calendar/addressbook

Note, that only users with a FreedomBox login can access Radicale.



If you want to share a calendar with only some users, the simplest approach is to create an additional user-name for these users and to share that user-name and password with them.

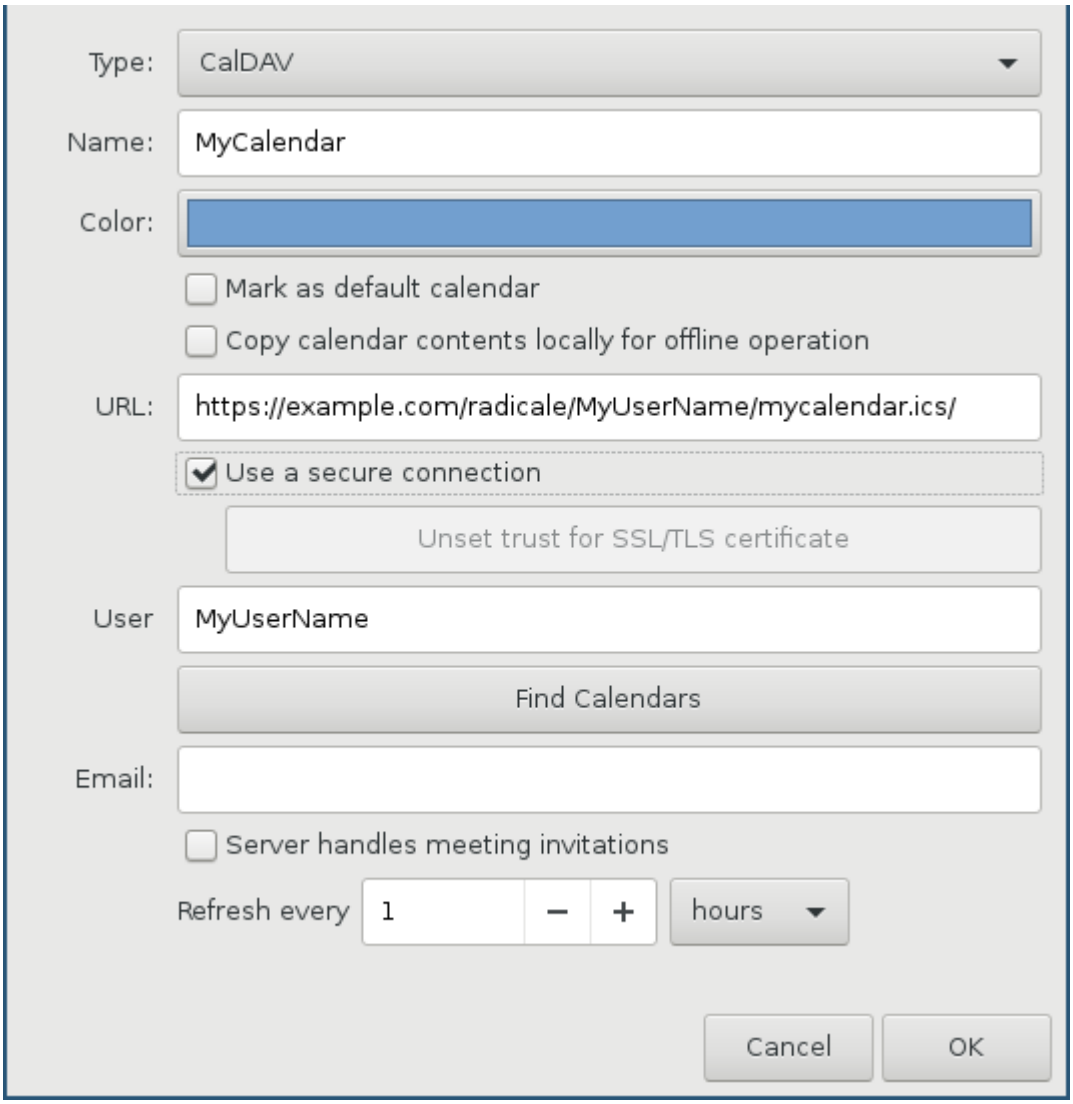
Radicale provides a basic web interface, which only supports creating new calendars and addressbooks. To add events or contacts, an external supported client application is needed.



- Creating addressbook/calendar using the web interface
  - Visit `https://`*IP-address-or-domain-for-your-server*`/radicale/`
  - Log in with your FreedomBox account
  - Select "Create new addressbook or calendar"
  - Provide a title and select the type
  - Optionally, provide a description or select a color
  - Click "Create"
  - The page will show the URL for your newly created addressbook or calendar

Now open your client application to create new calendar and address books that will use your FreedomBox and Radicale server. The Radicale website provides an overview of supported clients, but do *not* use the URLs described there; FreedomBox uses another setup, follow this manual. Below are the steps for two examples:

- Example of setup with Evolution client:
  - Calendar
    1. Create a new calendar
    2. For "Type," select "CalDAV"
    3. When "CalDAV" is selected, additional options will appear in the dialogue window.
    4. URL: `https://`*IP-address-or-domain-for-your-server*`/radicale/`*user*`/`*calendar-name*`.ics/`. Items in *italics* need to be changed to match your settings.
       * note the trailing `/` in the path, it is important.
    5. Enable "Use a secure connection."
    6. Name the calendar



- TODO/Tasks list: Adding a TODO/Tasks list is basically the same as a calendar.
- Contacts
  - Follow the same steps described above and replace CalDAV with WebDAV. The extension of the address book will be .vcf.

5.6.3 Synchronizing over Tor

In Plinth, setting up a calendar with Radicale over Tor is the same as over the clear net. Here is a short summary:

1. When logged in to Plinth over Tor, click on Radicale, and at the prompt provide your FreedomBox user name and password.
2. In the Radicale web interface, log in using your FreedomBox user name and password.
3. Click on "Create new address book or calendar", provide a title, select a type, and click "Create".
4. Save the URL, e.g., `https://`*ONION-ADDRESS-FOR-YOUR-SERVER*`.onion/radicale/`*USERNAME*`/`*CALENDAR-CODE*`/`. Items in *italics* need to be changed to match your settings.

These instructions are for Thunderbird/Lightning. Note that you will need to be connected to Tor with the Tor Browser Bundle.

1. Open Thunderbird, install the Torbirdy add-on, and restart Thunderbird. (This may not be necessary.)

2. In the Lightning interface, under Calendar/Home in the left panel right click with the mouse and select "New calendar".
3. Select the location of your calendar as "On the Network".
4. Select CalDAV and for the location copy the URL, e.g., `https://`*ONION-ADDRESS-FOR-YOUR-SERVER*`.onion/radical`*CODE*`/`. Items in *italics* need to be changed to match your settings.
5. Provide a name, etc. Click "Next". Your calendar is now syncing with your FreedomBox over Tor.
6. If you have not generated a certificate for your FreedomBox with "Let's Encrypt", you may need to select "Confirm Security Exception" when prompted.

5.6.4 Synchronizing with your Android phone

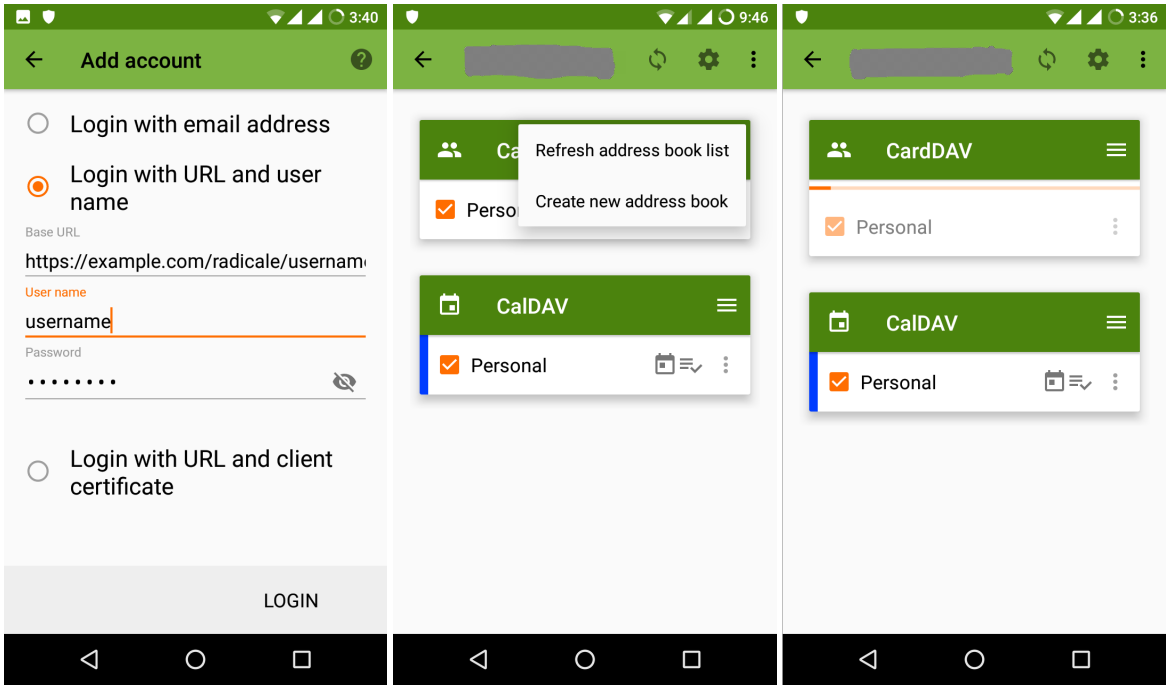
There are various Apps that allow integration with the Radicale server. This example uses DAVx5, which is available e.g. on F-Droid. If you intend to use ToDo-Lists as well, the compatible app OpenTasks has to be installed first.

Follow these steps for setting up your account with the Radicale server running on your FreedomBox.

1. Install DAVx5
2. Create a new account on DAVx5 by clicking on the floating + button.
3. Select the second option as shown in the first figure below and enter the base url as *`https://<your.freedombox.address>/radicale/username/`* (don't miss the `/` at the end). DAVx5 will be able to discover both CalDAV and WebDAV accounts for the user.
4. Follow this video from DAVx5 FAQ to learn how to migrate your existing contacts to Radicale.

Synchronizing contacts

1. Click on the hamburger menus of CalDAV and CardDAV and select either "Refresh ..." in case of existing accounts or "Create ..." in case of new accounts (see the second screenshot below).
2. Check the checkboxes for the address books and calendars you want to synchronize and click on the sync button in the header. (see the third screenshot below)



5.6.5 Advanced Users

5.6.5.1 Sharing resources

Above was shown an easy way to create a resource for a group of people by creating a dedicated account for all. Here will be described an alternative method where two users `User1` and `User2` are granted access to a calendar. This requires SSH-access to the FreedomBox.

1. create a file `/etc/radicale/rights`
   - ```
     [friends_calendar]
     user: ^(User1|User2)$
     collection: ^.*/calendar_of_my_friends.ics$
     permission: rw

     # Give write access to owners
     [owner-write]
     user: .+
     collection: ^%(login)s/.+$
     permission: rw
     ```
   - `[friends_calendar]` is just an identifier, can be any name.
   - The `[owner-write]` section makes sure that owners have access to their own files
2. edit file `/etc/radicale/config` and make the following changes in section `[rights)`
   - ```
     [rights]
     type = from_file
     file = /etc/radicale/rights
     ```
3. Restart the radicale server or the FreedomBox

5.6.5.2 Importing files

If you are using a contacts file exported from another service or application, it should be copied to: /var/lib/radicale/collections/*user*/*contact file name*.vcf.

5.6.6 Migrating from Radicale Version 1.x to Version 2.x

During the month of February 2019, radicale in Debian testing was upgraded from version 1.x to version 2.x. Version 2.x is a better version but is incompatible with data and configuration used with 1.x. Automatic upgrade mechanism in FreedomBox, handled by unattended-upgrades does not automatically upgrade radicale to version 2.x due to changes in configuration files. However, FreedomBox version 19.1, which is available on February 23rd, 2019 in testing will perform data and configuration migration to radicale version 2.x. Typical users require no action, this will happen automatically.

If for some reason, you need to manually run `apt dist-upgrade` on your machine, then radicale will be upgraded to 2.x and then FreedomBox will not be able to perform its upgrade (due to upstream project deciding to remove migration tools in radicale 2.x version). To avoid this situation, the following process is recommended if you wish to perform an upgrade.

```
sudo su -
apt hold radicale
apt dist-upgrade
apt unhold radicale
```

However, if you already happen to perform an upgrade to radicale 2.x without help from FreedomBox, you need to perform data and configuration migration yourself. Follow this procedure:

```
sudo su -
tar -cvzf /root/radicale_backup.tgz /var/lib/radicale/ /etc/radicale/ /etc/default/radicale
apt install -y python-radicale
python -m radicale --export-storage=/root/radicale-migration
cp -dpR /root/radicale-migration/collection-root /var/lib/radicale/collections/collection- ↩
    root/
(remove this directory if it already exists. Or perhaps merge the contents.)
chown -R radicale:radicale /var/lib/radicale/collections/collection-root/
apt remove -y python-radicale
if [ -f /etc/radicale/config.dpkg-dist ] ; then cp /etc/radicale/config.dpkg-dist /etc/ ↩
    radicale/config ; fi
if [ -f /etc/default/radicale.dpkg-dist ] ; then cp /etc/default/radicale.dpkg-dist /etc/ ↩
    default/radicale ; fi
(After FreedomBox 19.1 is available, goto FreedomBox web interface and set your preference  ↩
    for calendar sharing again, if it is not the default option, as it will have been lost.)
```

Notes:

- python-radicale is an old package from radicale 1.x version that is still available in testing. This is a hack to use the `--export-storage` feature that is responsible for data migration. This feature is not available in radicale 2.x unfortunately.
- Files ending with .dpkg-dist will exist only if you have chosen 'Keep your currently-installed version' when prompted for configuration file override during radicale 2.x upgrade. The above process will overwrite the old configuration with new fresh configuration. No changes are necessary to the two configuration files unless you have changed the setting for sharing calendars.
- Note that during the migration, your data is safe in /var/lib/radicale/collections directory. New data will be created and used in /var/lib/radicale/collections/collections-root/ directory.
- The tar command takes a backup your configuration and data in /root/radicale_backup.tgz in case you do something goes wrong and you want to undo the changes.

5.6.7 Troubleshooting

1. If you are using FreedomBox Pioneer Edition or installing FreedomBox on Debian Buster, then radicale may not be usable immediately after installation. This is due to a bug which has been fixed later. To overcome the problem, upgrade FreedomBox by clicking on 'Manual Update' from 'Updates' app. Otherwise, simply wait a day or two and let FreedomBox upgrade itself. After that install radicale. If radicale is already installed, disable and re-enable it after the update is completed. This will fix the problem and get radicale working properly.

5.7 Chat Server (ejabberd)

5.7.1 What is XMPP?

XMPP is a federated protocol for Instant Messaging. This means that users who have accounts on one server, can talk to users that are on another server. XMPP can also be used for voice and video calls, if supported by the clients.

With XMPP, there are two ways that conversations can be secured:

1. TLS: This secures the connection between the client and server, or between two servers. This should be supported by all clients and is highly recommended.
2. End-to-end: This secures the messages sent from one client to another, so that even the server cannot see the contents. The latest and most convenient protocol is called OMEMO, but it is only supported by a few clients. There is another protocol called OTR that may be supported by some clients that lack OMEMO support. Both clients must support the same protocol for it to work.

5.7.2 Setting the Domain Name

For XMPP to work, your FreedomBox needs to have a Domain Name that can be accessed over the public Internet. You can read more about obtaining a Domain Name in the Dynamic DNS section of this manual.

Once you have a Domain Name, you can tell your FreedomBox to use it by setting the Domain Name in the System Configuration.

- Note: After changing your Domain Name, the Chat Server (XMPP) page may show that the service is not running. After a minute or so, it should be up and running again.

Please note that PageKite does not support the XMPP protocol at this time.

5.7.3 Registering XMPP users through SSO

Currently, all users created through Plinth will be able to login to the XMPP server. You can add new users through the System Users and Groups module. It does not matter which Groups are selected for the new user.

5.7.4 Using the web client

After the XMPP module install completes, the JSXC web client for XMPP can be accessed at `https://<your freedombox>/pli`
It will automatically check the BOSH server connection to the configured domain name.

5.7.5 Using a desktop or mobile client

XMPP clients are available for various desktop and mobile platforms.

5.7.6 Port Forwarding

If your FreedomBox is behind a router, you will need to set up port forwarding on your router. You should forward the following ports for XMPP:

- TCP 5222 (client-to-server)
- TCP 5269 (server-to-server)

5.8 Chat Server (Matrix Synapse)

5.8.1 What is Matrix?

Matrix is an open standard for interoperable, decentralized, real-time communication over IP. Synapse is the reference implementation of a Matrix server. It can be used to setup instant messaging on FreedomBox to host large chat rooms, end-to-end encrypted communication and audio/video calls. Matrix Synapse is a federated application where chat rooms can exist on any server and users from any server in the federated network can join them. Learn more about Matrix.

Available since: version 0.14.0

5.8.2 How to access your Matrix Synapse server?

We recommend the Riot client to access the Matrix Synapse server. You can download Riot for desktops. Mobile applications for Android and iOS are available from their respective app stores.

5.8.3 Setting up Matrix Synapse on your FreedomBox

To enable Matrix, first navigate to the Chat Server (Matrix Synapse) page and install it. Matrix needs a valid domain name to be configured. After installation, you will be asked to configure it. You will be able to select a domain from a drop down menu of available domains. Domains are configured using System -> Configure page. After configuring a domain, you will see that the service is running. The service will be accessible on the configured FreedomBox domain. Currently, you will not be able to change the domain once is it configured.

Your router has to be configured to forward port 8448.

All the registered users of your FreedomBox will have their Matrix IDs as `@username:domain`. If public registration is enabled, also your chosen client can be used to register a user account.

5.8.4 Federating with other Matrix instances

You will be able to interact with any other person running another Matrix instance. This is done by simply starting a conversation with them using their matrix ID which is of the format `@their-username:their-domain`. You can also join rooms which are in another server and have audio/video calls with contacts on other server.

5.8.5 Memory usage

The Synapse reference server implemented in Python is known to be quite RAM hungry, especially when loading large rooms with thousands of members like #matrix:matrix.org. It is recommended to avoid joining such rooms if your FreedomBox device only has 1 GiB RAM or less. Rooms with up to a hundred members should be safe to join. The Matrix team is working on a new implementation of the Matrix server written in Go called Dendrite which might perform better in low-memory environments.

Some large public rooms in the Matrix network are also available as IRC channels (e.g. #freedombox:matrix.org is also available as #freedombox on irc.debian.org). It is better to use IRC instead of Matrix for such large rooms. You can join the IRC channels using Quassel.

5.8.6 Advanced usage

1. If you wish to create a large number of users on your Matrix Synapse server, use the following commands on a remote shell as root user:

   ```
   cat /dev/urandom | tr -dc 'a-zA-Z0-9' | fold -w 32 | head -n 1 | sed "s+^+ ↩
       registration_shared_secret: +" > /etc/matrix-synapse/conf.d/ ↩
       registration_shared_secret.yaml
   chmod 600 /etc/matrix-synapse/conf.d/registration_shared_secret.yaml
   chown matrix-synapse:nogroup /etc/matrix-synapse/conf.d/registration_shared_secret. ↩
       yaml
   systemctl restart matrix-synapse
   register_new_matrix_user -c /etc/matrix-synapse/conf.d/registration_shared_secret. ↩
       yaml
   ```

2. If you wish to see the list of users registered in Matrix Synapse, the following as root user:

   ```
   apt install sqlite3
   echo 'select name from users' | sqlite3 /var/lib/matrix-synapse/homeserver.db
   ```

3. If you wish to create a community in Matrix Synapse, a Matrix user with server admin privileges is needed. In order to grant such privileges to `username` run the following commands as root user:

   ```
   sudo apt install sqlite3
   echo "UPDATE users SET admin=1 WHERE name='@username:domainname'" | sudo sqlite3 /var ↩
       /lib/matrix-synapse/homeserver.db
   ```

5.9 Email Client (Roundcube)

5.9.1 What is Roundcube?

Roundcube is a browser-based multilingual email client with an application-like user interface. Roundcube is using the Internet Message Access Protocol (IMAP) to access e-mail on a remote mail server. It supports MIME to send files, and provides particularly address book, folder management, message searching and spell checking.

5.9.2 Using Roundcube

After Roundcube is installed, it can be accessed at `https://<your freedombox>/roundcube`. Enter your username and password. The username for many mail services will be the full email address such as *exampleuser@example.org* and not just the username like *exampleuser*. Enter the address of your email service's IMAP server address in the *Server* field. You can try providing your domain name here such as *example.org* for email address *exampleuser@example.org* and if this does not work, consult your email provider's documentation for the address of the IMAP server. Using encrypted connection to your IMAP server is strongly recommended. To do this, prepend 'imaps://' at the beginning of your IMAP server address. For example, *imaps://imap.example.org*.

5.9.3 Using Gmail with Roundcube

If you wish to use Roundcube with your Gmail account, you need to first enable support for password based login in your Google account preferences. This is because Gmail won't allow applications to login with a password by default. To do this, visit Google Account preferences and enable *Less Secure Apps*. After this, login to Roundcube by providing your Gmail address as *Username*, your password and in the server field use *imaps://imap.gmail.com*.

5.10 File Sharing (Coquelicot)

5.10.1 About Coquelicot

Coquelicot is a "one-click" file sharing web application with a focus on protecting users' privacy. The basic principle is simple: users can upload a file to the server, in return they get a unique URL which can be shared with others in order to download the file. A download password can be defined.

After the upload you get a unique link that can be shared to your partners in order to

Read more about Coquelicot at the Coquelicot README

Available since: version 0.24.0

5.10.2 When to use Coquelicot

Coquelicot is best used to quickly share a single file. If you want to share a folder,

1. for a single use, compress the folder and share it over Coquelicot
2. which must be kept synchronized between computers, use Syncthing instead

Coquelicot can only provide a reasonable degree of privacy. If anonymity is required, you should consider using the desktop application Onionshare instead.

Since Coquelicot fully uploads the file to the server, your FreedomBox will incur both upload and download bandwidth costs. For very large files, consider sharing them using BitTorrent by creating a private torrent file. If anonymity is required, use Onionshare. It is P2P and doesn't require a server.

5.10.3 Coquelicot on FreedomBox

With Coquelicot installed, you can upload files to your FreedomBox server and privately share them.

Post installation, the Coquelicot page offers two settings.

1. **Upload Password**: Coquelicot on FreedomBox is currently configured to use simple password authentication for ease of use. Remember that it's one global password for this Coquelicot instance and not your user password for FreedomBox. You need not remember this password. You can set a new one from the Plinth interface anytime.
2. **Maximum File Size**: You can alter the maximum size of the file that can be transferred through Coquelicot using this setting. The size is in Mebibytes. The maximum file size is only limited by the disk size of your FreedomBox.

5.10.4 Privacy

Someone monitoring your network traffic might find out that some file is being transferred through your FreedomBox and also possibly its size, but will not know the file name. Coquelicot encrypts files on the server and also fills the file contents with 0s when deleting them. This eliminates the risk of file contents being revealed in the event of your FreedomBox being confiscated or stolen. The real risk to mitigate here is a third-party also downloading your file along with the intended recipient.

5.10.4.1 Sharing over instant messengers

Some instant messengers which have previews for websites might download your file in order to show a preview in the conversation. If you set the option of one-time download on a file, you might notice that the one download will be used up by the instant messenger. If sharing over such messengers, please use a download password in combination with a one-time download option.

5.10.4.2 Sharing download links privately

It is recommended to share your file download links and download passwords over encrypted channels. You can simply avoid all the above problems with instant messenger previews by using instant messengers that support encrypted conversations like Riot with Matrix Synapse or XMPP (ejabberd server on FreedomBox) with clients that support end-to-end encryption. Send the download link and the download password in two separate messages (helps if your messenger supports perfect forward secrecy like XMPP with OTR). You can also share your links over PGP-encrypted email using Thunderbird.

5.11 File Sharing (MLDonkey)

5.11.1 What is MLDonkey?

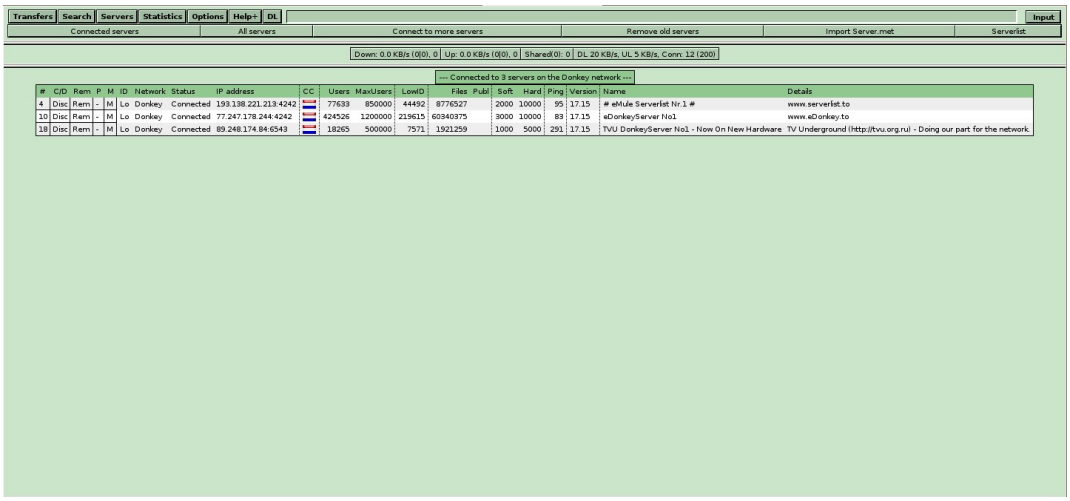
MLDonkey is an open-source, multi-protocol, peer-to-peer file sharing application that runs as a back-end server application on many platforms. It can be controlled through a user interface provided by one of many separate front-ends, including a Web interface, telnet interface and over a dozen native client programs.

Originally a Linux client for the eDonkey protocol, it now runs on many flavors of Unix-like, OS X, Microsoft Windows and MorphOS and supports numerous peer-to-peer protocols including ED2K (and Kademlia and Overnet), BitTorrent, DC++ and more.

Read more about MLDonkey at the MLDonkey Project Wiki

Available since: version 0.48.0

5.11.2 Screenshot



5.11.3 Using MLDonkey Web Interface

After installing MLDonkey, its web interface can be accessed from FreedomBox at `https://<your freedombox>/mldonkey`. Users belonging to the *ed2k* and *admin* groups can access this web interface.

5.11.4 Using Desktop/Mobile Interface

Many desktop and mobile applications can be used to control MLDonkey. MLDonkey server will always be running on FreedomBox. It will download files (or upload them) and store them on FreedomBox even when your local machine is not running or connected to MLDonkey on FreedomBox. Only users of *admin* group can access MLDonkey on FreedomBox using desktop or mobile clients. This is due to restrictions on which group of users have SSH access into FreedomBox.

1. Create an admin user or use an existing admin user.
2. On your desktop machine, open a terminal and run the following command. It is recommended that you configure and use SSH keys instead of passwords for the this step.

   ```
   $ ssh -L 4001:localhost:4001 -N exampleuser@example.freedombox.rocks
   ```

3. Start the GUI application and then connect it to MLDonkey as if MLDonkey is running on the local desktop machine. After you are done, terminate the SSH command by pressing Control-C.

See MLDonkey documentation for SSH Tunnel for more information.

5.12 File Synchronization (Syncthing)

With *Syncthing* installed on your FreedomBox, you can synchronize content from other devices to your FreedomBox and vice-versa. For example, you can keep the photos taken on your mobile phone synchronized to your FreedomBox.

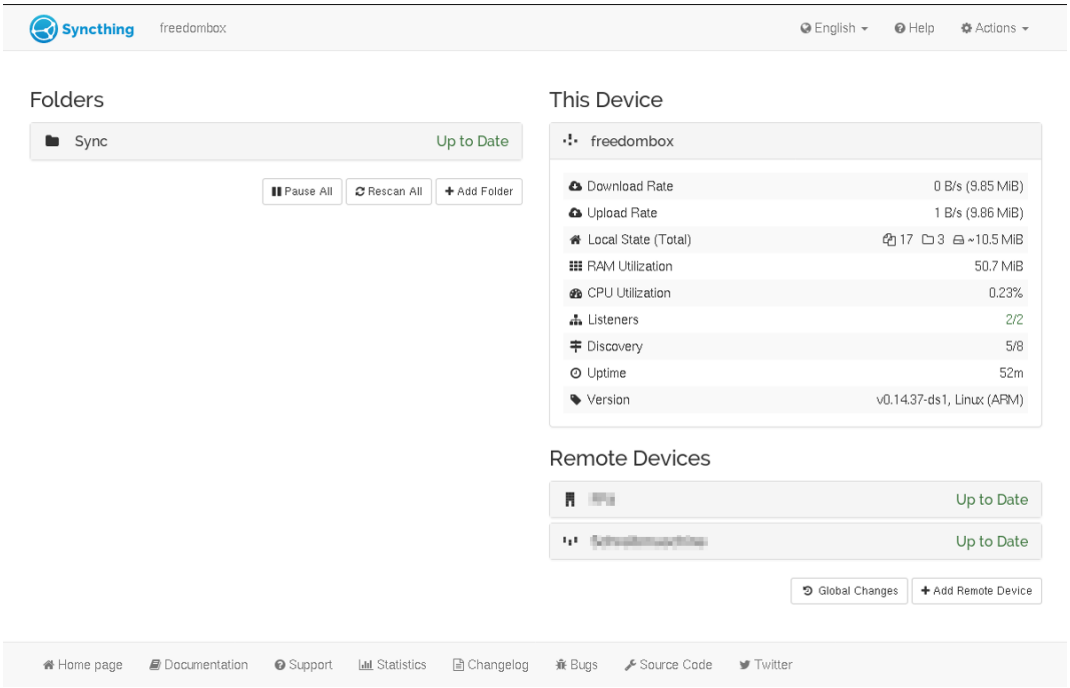
Available since version: 0.14

Users should keep in mind that Syncthing is a peer-to-peer synchronization solution, not a client-server one. This means that the FreedomBox isn't really the server and your other devices clients. They're all devices from Syncthing's perspective. You can use Syncthing to synchronize your files between any of your devices. The advantage that FreedomBox provides is that it is a server that's always running. Suppose you want your photos on your phone to be synchronized to your laptop, if you simply sync the photos to the FreedomBox, the laptop can get them from the FreedomBox whenever it comes online the next time. You don't have to be worried about your other devices being online for synchronization. If your FreedomBox is one of the devices set up

with your Syncthing shared folder, you can rest assured that your other devices will eventually get the latest files once they come online.

After installation follow the instructions in the getting started of the Syncthing project. Syncthing allows individual folders to be selectively shared with other devices. Devices must be paired up before sharing by scanning QR codes or entering the device ids manually. Syncthing has a discovery service for easily identifying the other devices on the same network having Syncthing installed.

In order to access to the web client of the Syncthing instance running on your FreedomBox, use the path */syncthing*. This web client is currently only accessible to the users of the FreedomBox that have administrator privileges, though it might be accessible to all FreedomBox users in a future release.



Syncthing has android apps available on the F-Droid and Google Play app stores. Cross-platform desktop apps are also available.

To learn more about Syncthing, please visit their official website and documentation.

5.12.1 Synchronizing over Tor

Syncthing should automatically sync with your FreedomBox even if it is only accessible as a Tor Onion Service.

If you would like to proxy your Syncthing client over Tor, set the `all_proxy` environment variable:

```
$ all_proxy=socks5://localhost:9050 syncthing
```

For more information, see the Syncthing documentation on using proxies.

5.12.2 Avoiding Syncthing Relays

Syncthing uses dynamic connections by default to connect with other peers. This means that if you are synchronizing over the Internet, the data might have to go through public Syncthing relays to reach your devices. This doesn't take advantage of the fact that your FreedomBox has a public IP address.

When adding your FreedomBox as a device in other Syncthing clients, set the address like "tcp://<my.freedombox.domain>" instead of "dynamic". This allows your Syncthing peers to directly connect to your FreedomBox avoiding the need for relays. It also allows for fast on-demand syncing if you don't want to keep Syncthing running all the time on your mobile devices.

5.12.3 Using Syncthing with other applications

5.12.3.1 Password Manager

Password managers that store their databases in files are suitable for synchronization using Syncthing. The following example describes using a free password manager called KeePassXC in combination with Syncthing to serve as a replacement for proprietary password managers that store your passwords in the cloud.

KeePassXC stores usernames, passwords etc. in files have the .kdbx extension. These kdbx files can be stored in a Syncthing shared folder to keep them synchronized on multiple machines. Free software applications which can read this file format are available for both desktop and mobile. You typically have to just point the application at the .kdbx file and enter the master password to access your stored credentials. For example, the same kdbx file can be accessed by using KeePassXC on desktop and KeePassDX on Android. KeePassXC can also be used to fill credentials into login fields in the browser by installing a browser extension.

5.13 File Synchronization (Samba)

Samba lets you have a shared folder over the local network that can be used from multiple computers running different operating systems. We will henceforth refer to these shared folders as "shares".

You can have a personal folder shared between your own devices (Home share), a folder shared with a trusted group (Group share) or one that is shared with every device on the network (Open share).

Samba lets you to treat a share as if it's a local folder on your computer. However, shares are available only on the local network.

To learn more about Samba, please refer to the user documentation on their wiki.

Available since version: 19.22

5.13.1 Using Samba

After installation, you can choose which disks to use for sharing. Enabled shares are accessible in the file manager on your computer at location \\freedombox (on Windows) or smb://freedombox.local (on Linux and Mac). There are three types of shares you can choose from:

Open share - accessible to everyone in your local network.

Group share - accessible only to FreedomBox users who are in the *freedombox-share* group.

Home share - every user in the *freedombox-share group* can have their own private space.

5.13.1.1 On Android

To access Samba shares on an Android device, install "Android Samba Client" from F-Droid or Google Play. Enter *smb://freedombox.loc*
as the share path in the app. Your shared folders should then be visible in the file manager app.

5.13.2 Integration with other apps

Transmission app on FreedomBox provides a setting to allow downloads to be saved directly to a Samba share.

5.13.3 Comparison with other apps

5.13.3.1 Syncthing

Syncthing maintains a copy of the shared folder on each device that it is shared with. Samba maintains only one copy on your FreedomBox device.

Syncthing can synchronize your shared folders between devices over the Internet. Samba shares are only available on the local network.

Since Syncthing is primarily a synchronization solution, it has features like conflict resolution and versioning. Samba has only copy of the file, so it doesn't need such features. For example, if two people are editing a spreadsheet stored on a Samba share, the last one to save the file wins.

5.14 Gobby Server (infinoted)

infinoted is a server for Gobby, a collaborative text editor.

To use it, download Gobby, desktop client and install it. Then start Gobby and select "Connect to Server" and enter your FreedomBox's domain name.

5.14.1 Port Forwarding

If your FreedomBox is behind a router, you will need to set up port forwarding on your router. You should forward the following ports for infinoted:

- TCP 6523

5.15 IRC Client (Quassel)

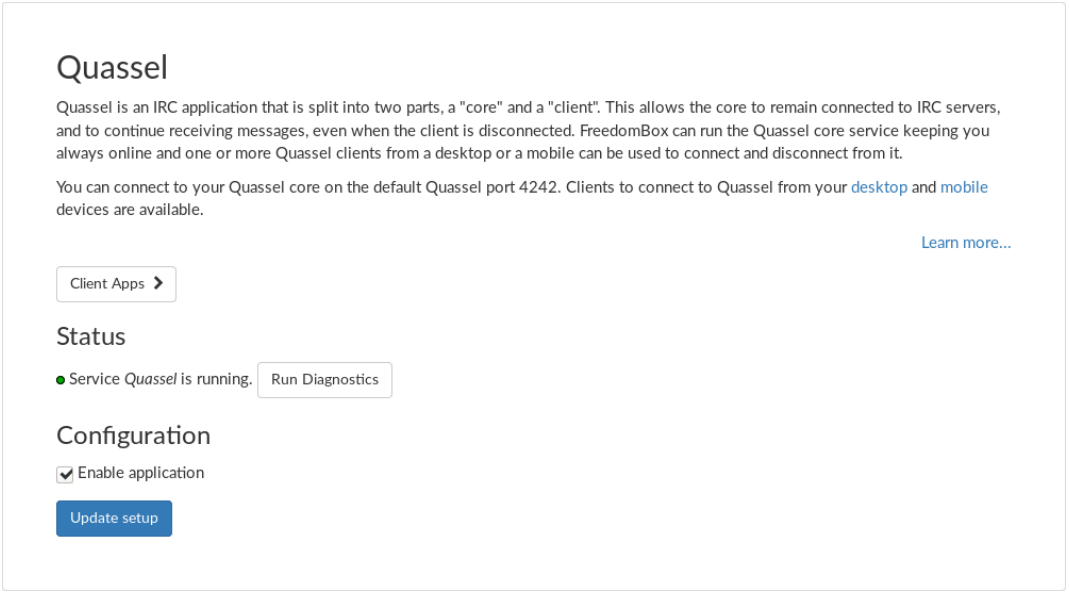
Quassel is an IRC application that is split into two parts, a "core" and a "client". This allows the core to remain connected to IRC servers, and to continue receiving messages, even when the client is disconnected. FreedomBox can run the Quassel core service keeping you always online and one or more Quassel clients from a desktop or a mobile device can be used to connect and disconnect from it.

5.15.1 Why run Quassel?

Many discussions about FreedomBox are being done on the IRC-Channel `irc://irc.debian.org/freedombox`. If your FreedomBox is running *Quassel*, it will collect all discussions while you are away, such as responses to your questions. Remember, the FreedomBox project is a worldwide project with people from nearly every time zone. You use your *client* to connect to the *Quassel* core to read and respond whenever you have time and are available.

5.15.2 How to setup Quassel?

- Within Plinth
  1. select *Applications*
  2. go to *IRC Client (Quassel)* and
  3. install the application and make sure it is enabled

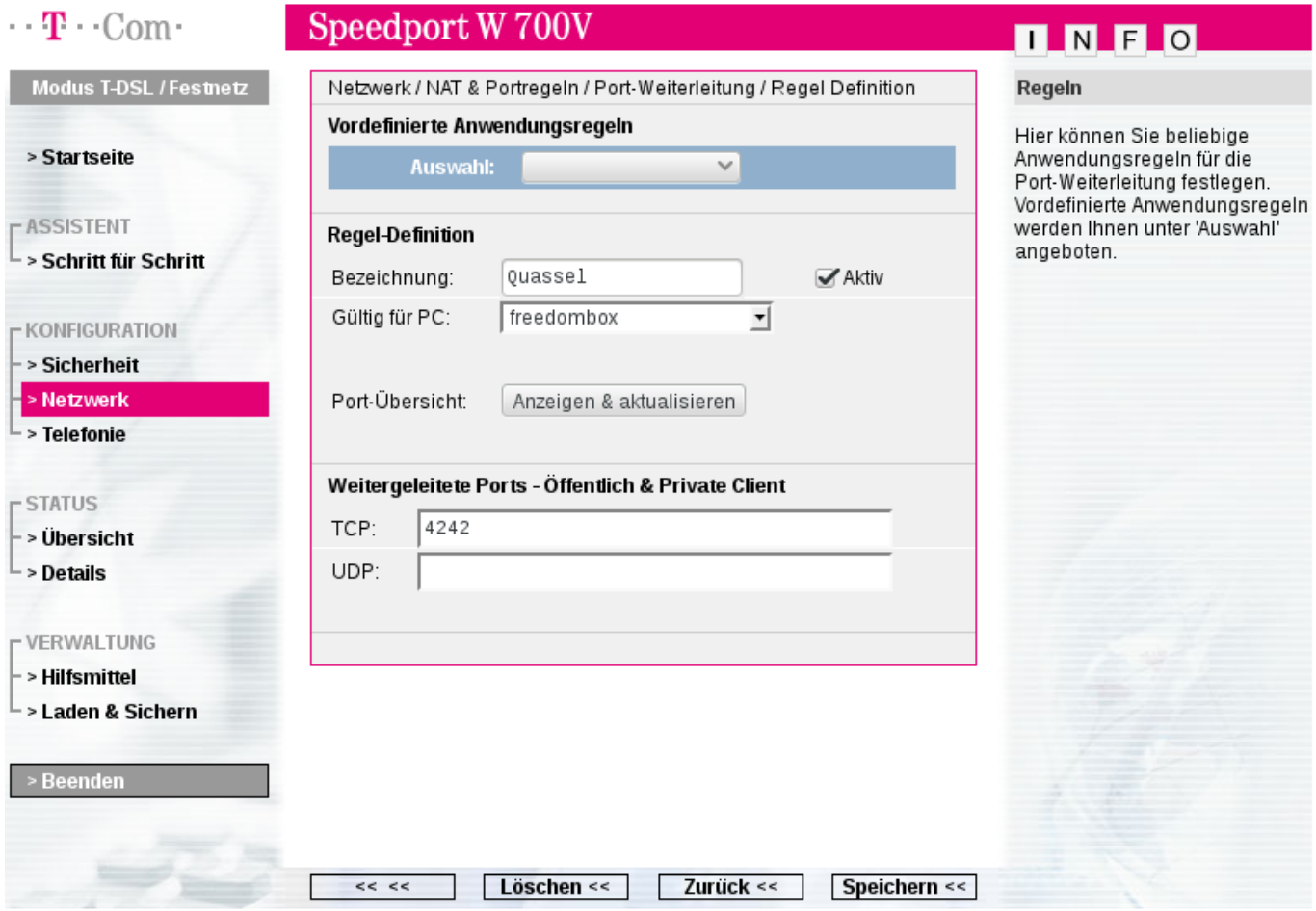


  4. now your Quassel core is running

5.15.3 Port Forwarding

If your FreedomBox is behind a router, you will need to set up port forwarding on your router. You should forward the following ports for Quassel:

- TCP 4242
- Example configuration in router:



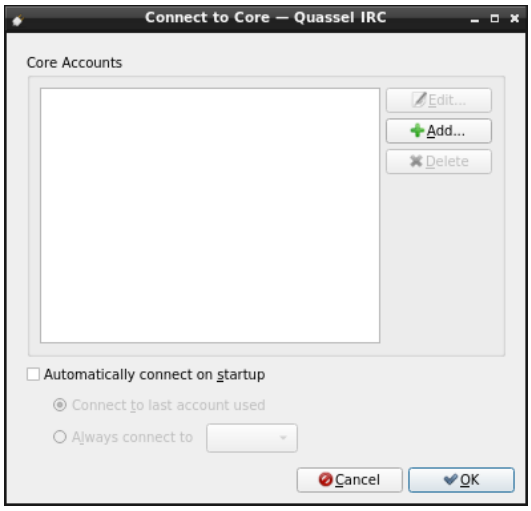
5.15.4 Clients

Clients to connect to Quassel from your desktop and mobile devices are available.

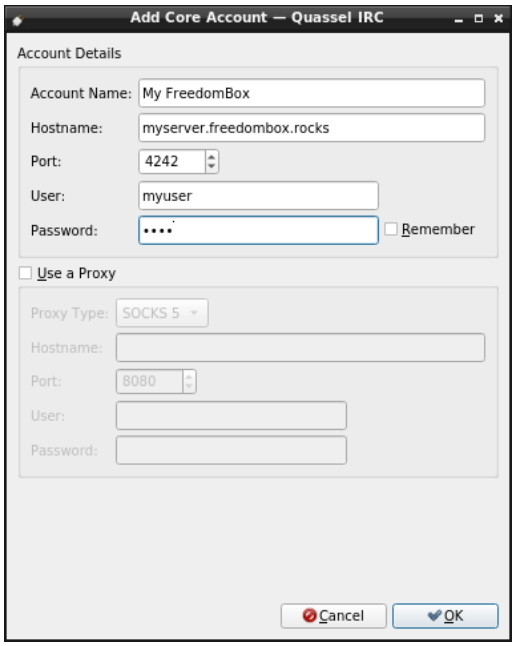
5.15.4.1 Desktop

In a Debian system, you can e.g. use quassel-client. The following steps describe how to connect Quassel Client with Quassel Core running on a FreedomBox. The first time you do this connection, Quassel Core will be initialized too.

1. Launch Quassel Client. You will be greeted with a wizard to `Connect to Core`.

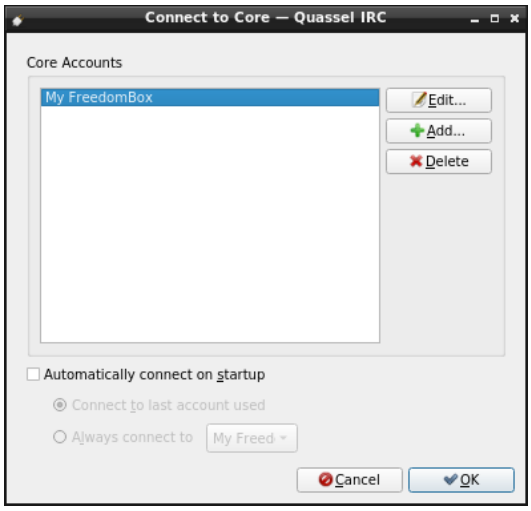


2. Click the `Add` button to launch `Add Core Account` dialog.

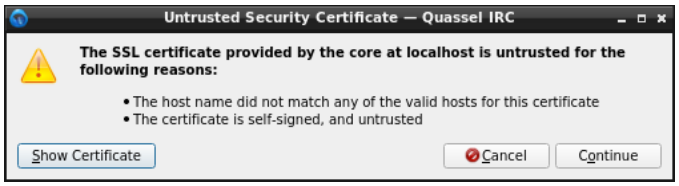


3. Fill any value in the `Account Name` field. Fill proper DNS hostname of your FreedomBox in `Hostname` filed. `Port` field must have the value `4242`. Provide the username and password of the account you wish to create to connect to the Quassel Core in the `User` and `Password` fields. Choose `Remember` if don't wish to be prompted for a password every time you launch Quassel client.

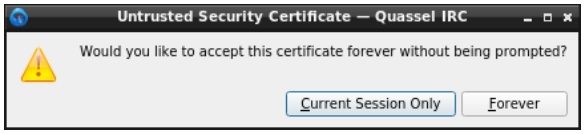
4. After pressing `OK` in the `Add Core Account` dialog, you should see the core account in the `Connect to Core` dialog.

- 

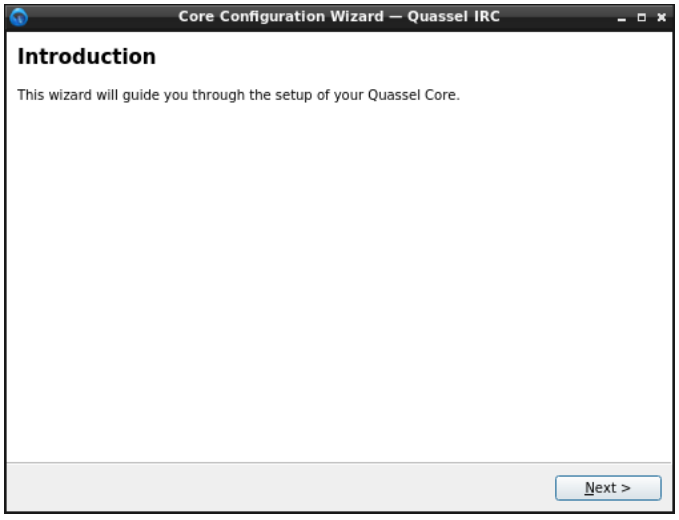
5. Select the newly created core account and select `OK` to connect to it.
6. If this is the first time you are connecting to this core. You will see an `Untrusted Security Certificate` warning and need to accept the server certificate.

- 

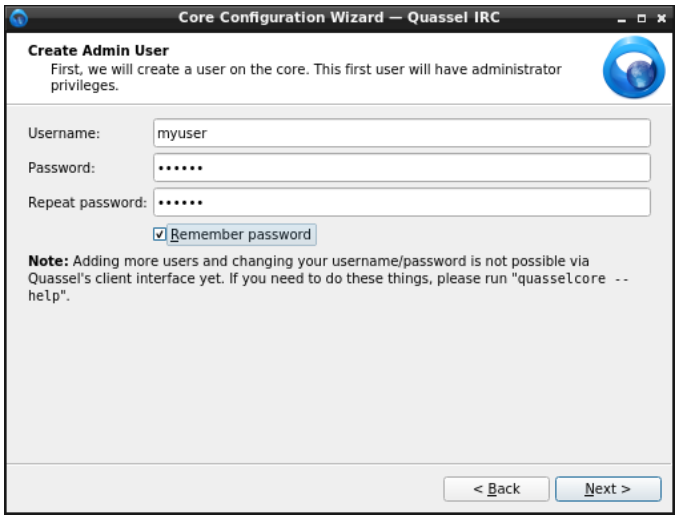
7. Select `Continue`. Then you will be asked if you wish to accept the certificate permanently. Select `Forever`.

- 

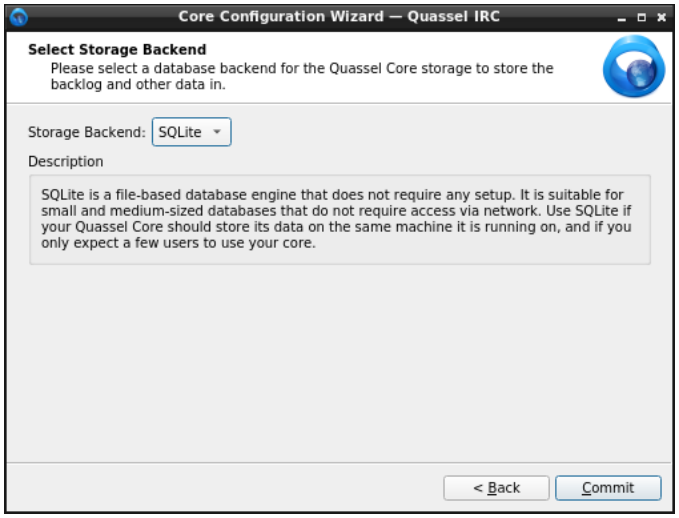
8. If this Quassel Core has not been connected to before, you will then see a `Core Configuration Wizard`. Select `Next`.

- 

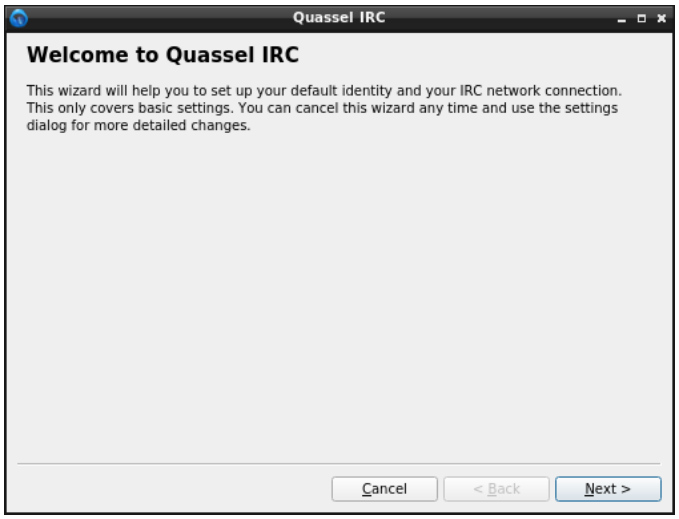
9. In the `Create Admin User` page, enter the username and password you have used earlier to create the core connection. Select `Remember password` to remember this password for future sessions. Click `Next`.



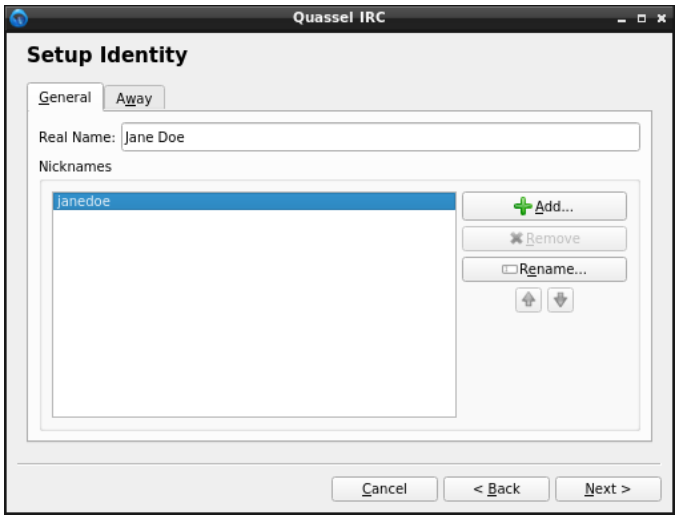
10. In the `Select Storage Backend` page, select `SQLite` and click `Commit`.



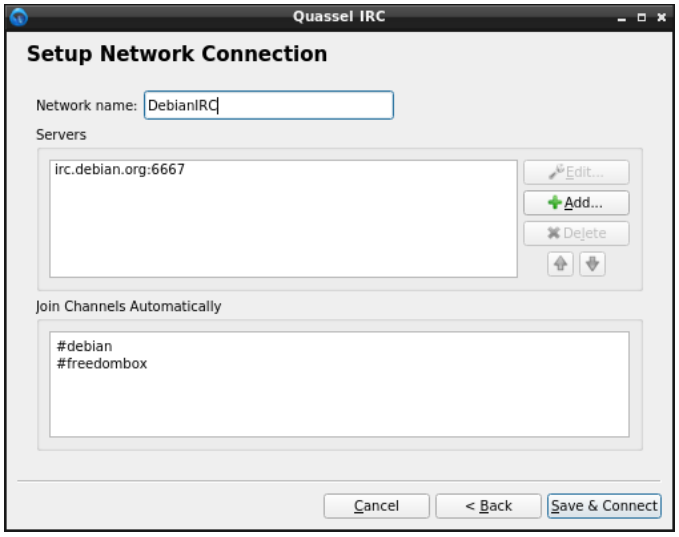
11. The core configuration is then complete and you will see a `Quassel IRC` wizard to configure your IRC connections. Click `Next`.



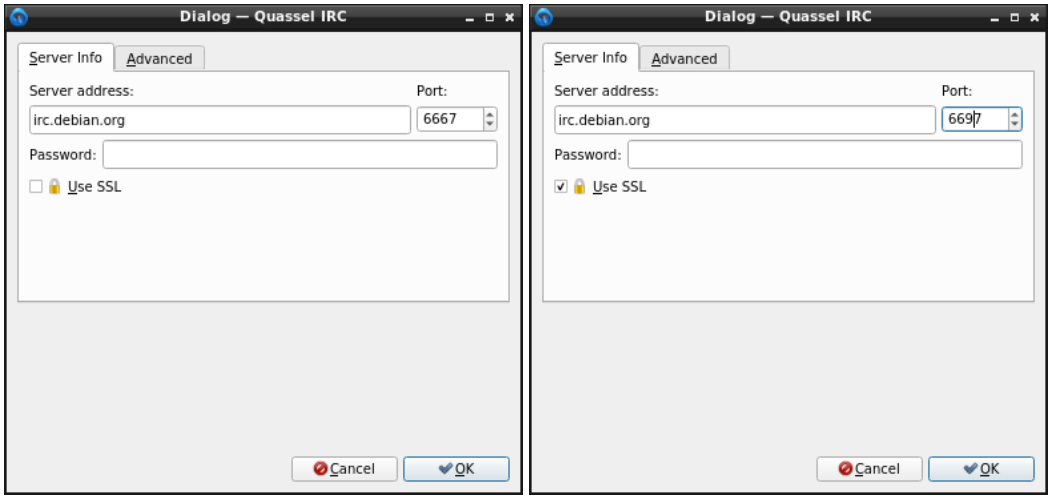
12. In `Setup Identity` page next, provide a name and multiple nicknames. This is how you present yourself to other users on IRC. It is not necessary to give your real world name. Multiple nicknames are useful as fallback nicknames when the first nickname can't be used for some reason. After providing the information click `Next`.



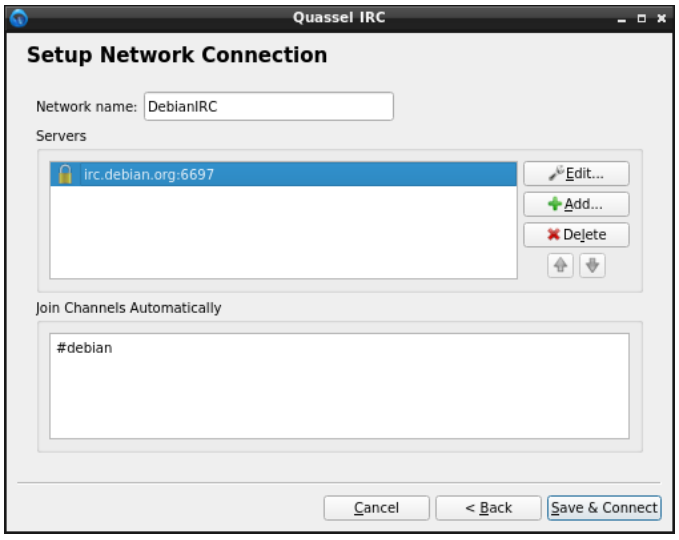
13. In `Setup Network Connection` page next, provide a network name of your choice. Next provide a list of servers to which Quassel Core should connect to in order to join this IRC network (such as irc.debian.org:6667).



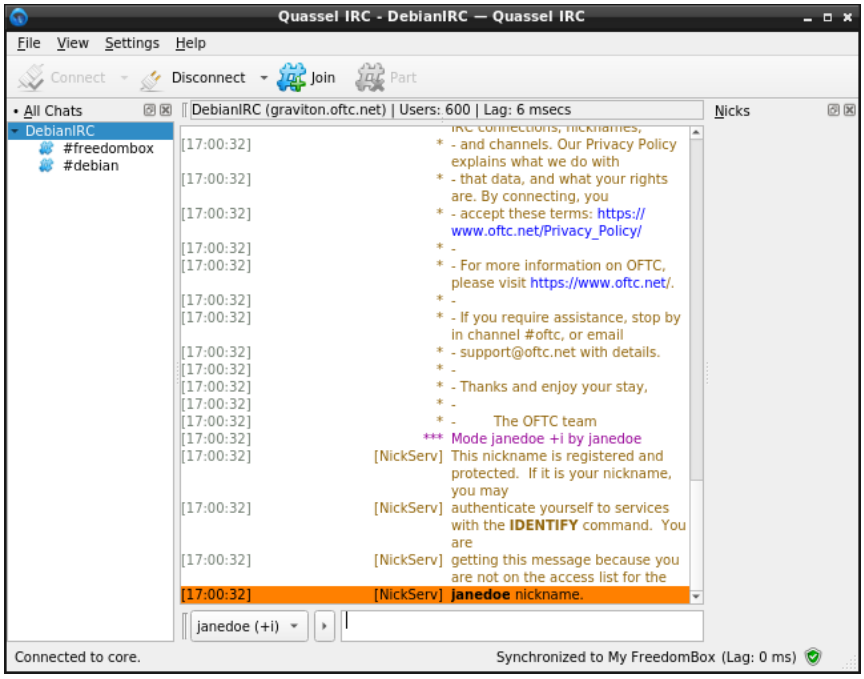
14. Select the server in the servers list and click `Edit`. In the `Server Info` dialog, set the port `6697` (consult your network's documentation for actual list of servers and their secure ports) and click `Use SSL`. Click `OK`. This is to ensure that communication between your FreedomBox and the IRC network server is encrypted.



15. Back in the `Setup Network Connection` dialog, provide a list of IRC channels (such as #freedombox) to join upon connecting to the network. Click `Save & Connect`.



16. You should connect to the network and see the list of channels you have joined on the `All Chats` pane on the left of the Quassel Client main window.

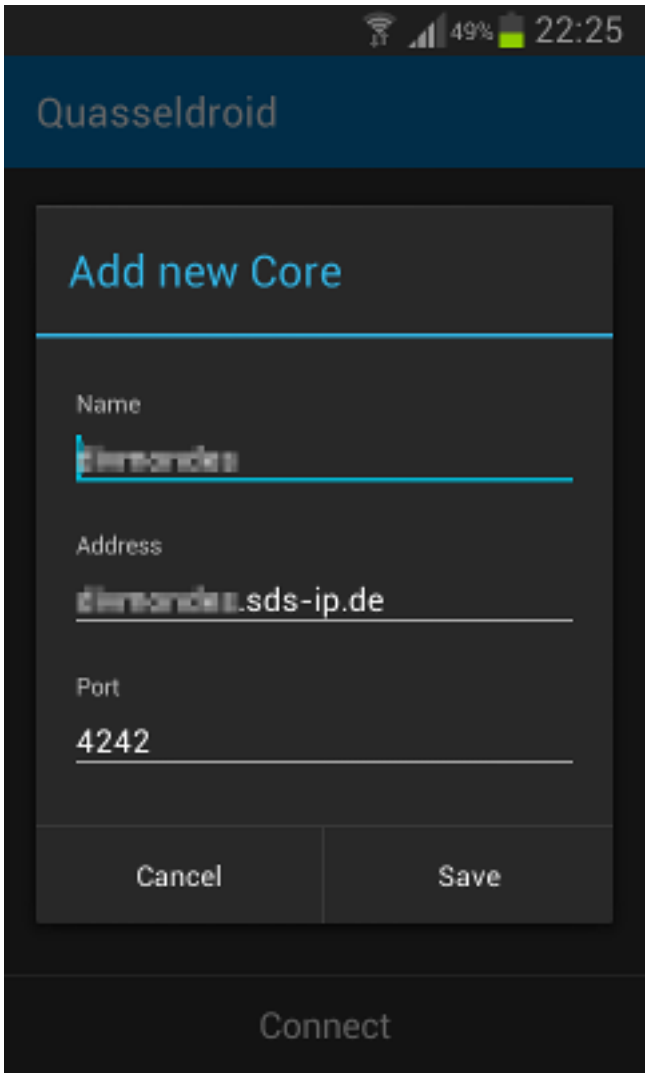


17. Select a channel and start seeing messages from others in the channel and send your own messages.

5.15.4.2 Android

For Android devices you may use e.g. *Quasseldroid* from F-Droid

- enter core, username etc. as above

– 

By the way, the German verb *quasseln* means *talking a lot*, to *jabber*.

5.16 MiniDLNA

MiniDLNA is a media server with the aim to be compliant with DLNA/UPnP clients.

5.16.1 What is UPnP/DLNA?

Universal plug & play is a set of networking protocols that allow devices within a network such as PCs, TVs, printers etc. to seamlessly discover each other and establish communication for data sharing. It is zero configuration protocol and requires only a media server and a media player that are compliant with the protocol.

DLNA is derived from UPnP as a form of standardizing media interoperability. It forms a standard/certification which many consumer electronics conform to.

5.16.2 Setting up MiniDLNA on your FreedomBox.

To install/enable the media server you need to navigate at MiniDLNA page and enable it. The application is intended to be available in the internal (home) network and therefore it requires a network interface configured for internal traffic.

After installation a web page becomes available on `https://<your-freedombox>/_minidlna`. It includes information for how many files the server is detecting, how many connections exist etc. This is very useful if plugging external disks with media to check if the new media files are detected properly. If that is not happening, disabling and enabling the server will fix it.

5.16.3 File systems for external drives.

If using an external drive that is used also from a Windows system the preferred filesystem should be NTFS. NTFS will keep Linux file permissions and UTF8 encoding for file names. This is useful if file names are in your language.

5.16.4 External links

http://minidlna.sourceforge.net/ https://en.wikipedia.org/wiki/Digital_Living_Network_Alliance

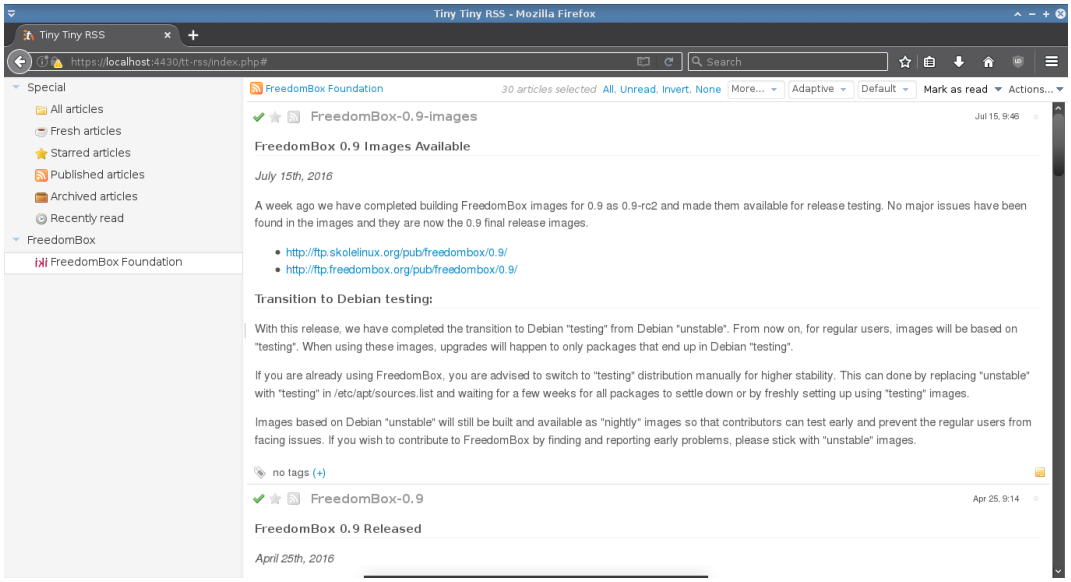
5.17 News Feed Reader (Tiny Tiny RSS)

Tiny Tiny RSS is a news feed (RSS/Atom) reader and aggregator, designed to allow reading news from any location, while feeling as close to a real desktop application as possible.

Any user created through FreedomBox web interface will be able to login and use this app. Each user has their own feeds, state and preferences.

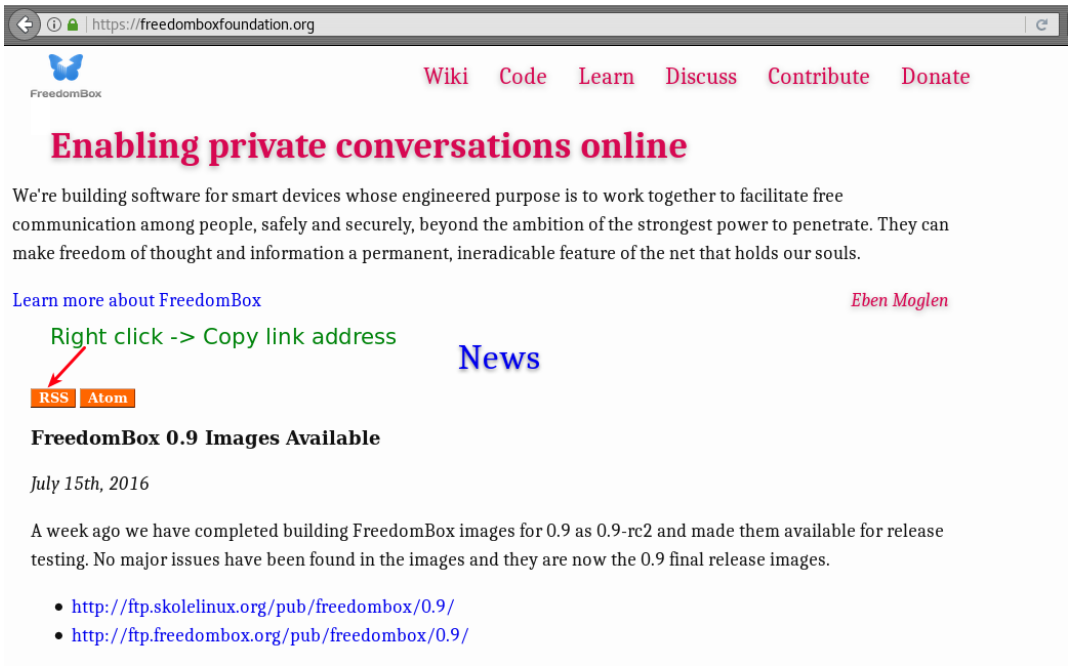
5.17.1 Using the Web Interface

When enabled, Tiny Tiny RSS will be available from */tt-rss* path on the web server. Any user created through Plinth will be able to login and use this app.

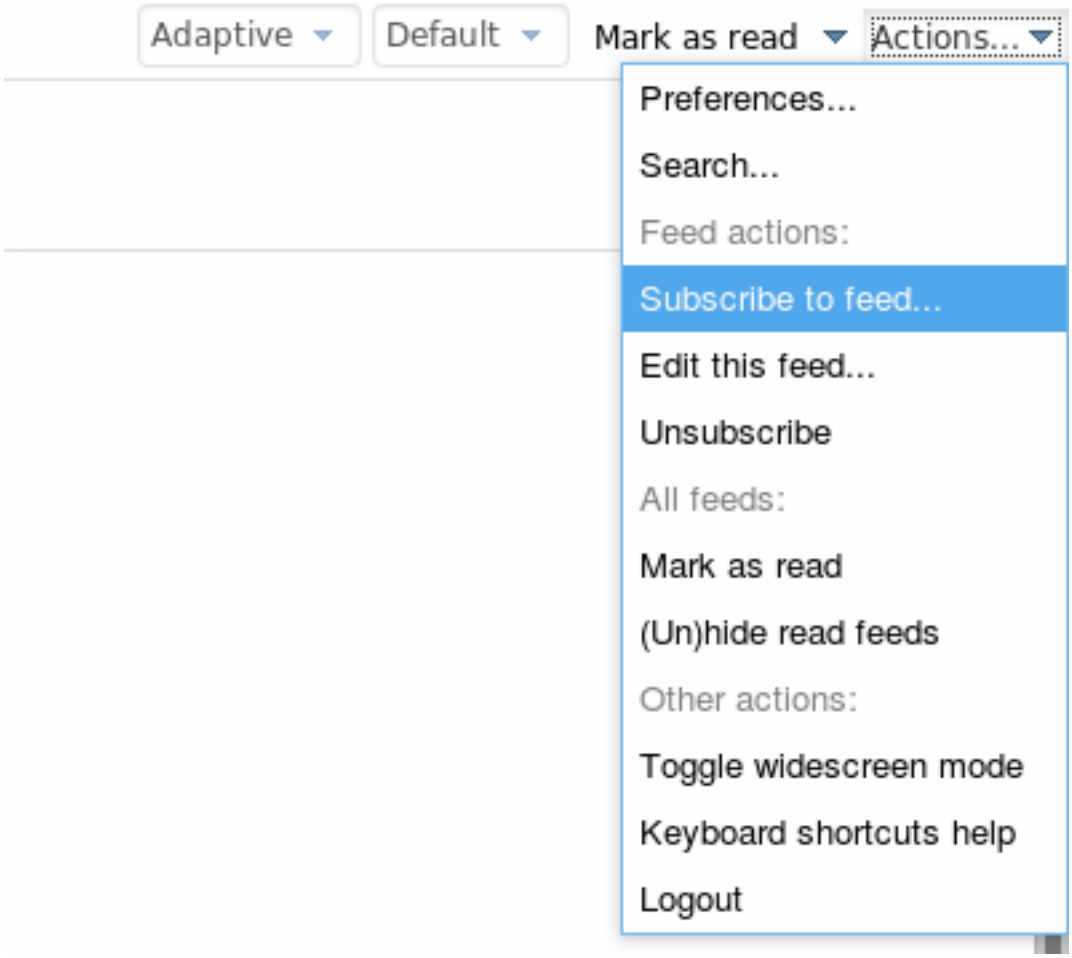


5.17.1.1 Adding a new feed

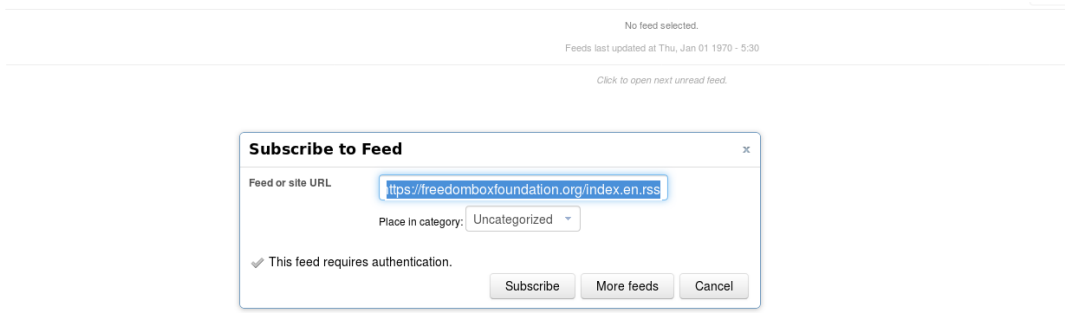
1. Go to the website you want the RSS feed for and copy the RSS/Atom feed link from it.



2. Select "Subscribe to feed.." from the Actions dropdown.



3. In the dialog box that appears, paste the URL for copied in step 1 and click the **Subscribe** button.

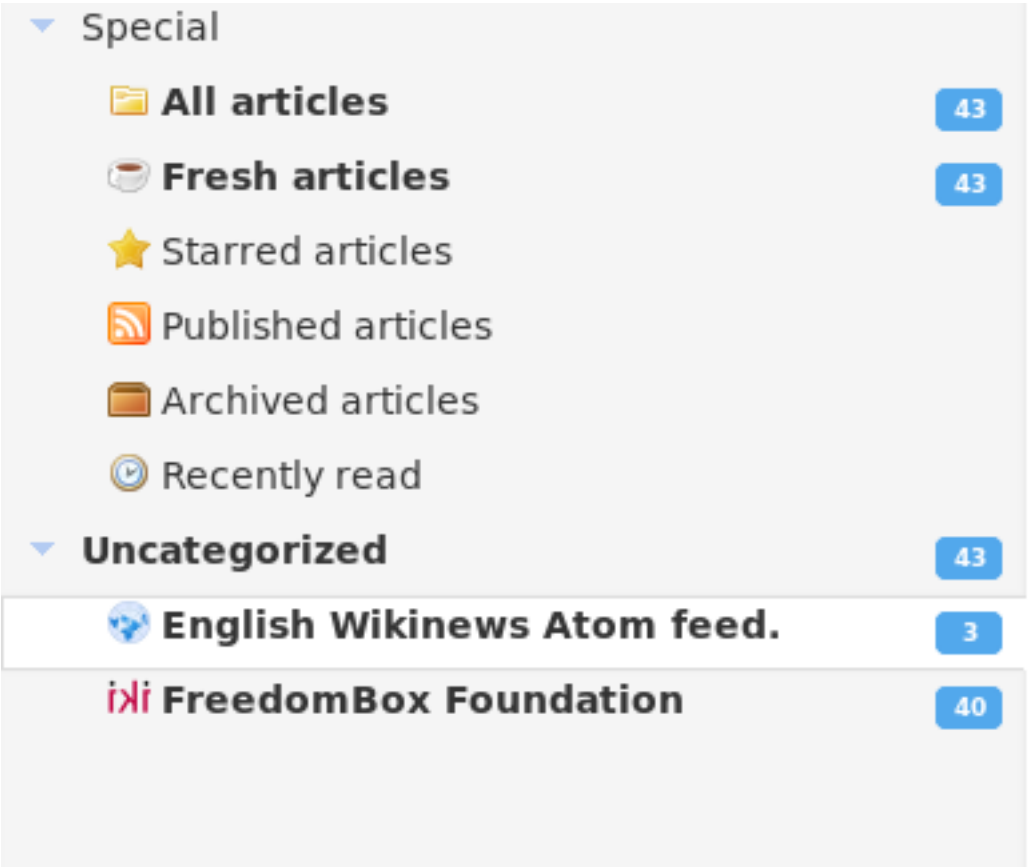


Give the application a minute to fetch the feeds after clicking Subscribe.

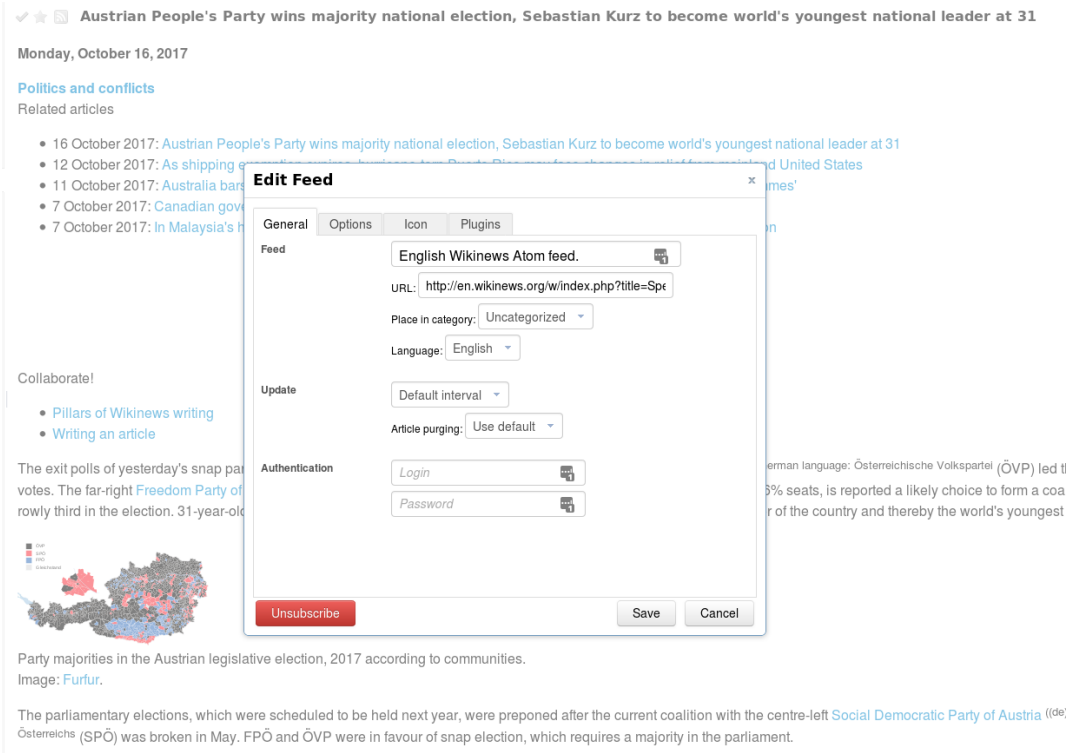
In some websites, the RSS feeds button isn't clearly visible. In that case, you can simply paste the website URL into the Subscribe dialog (step 3) and let TT-RSS automatically detect the RSS feeds on the page.

You can try this now with the homepage of WikiNews

As you can see in the image below, TT-RSS detected and added the Atom feed of WikiNews to our list of feeds.



If you don't want to keep this feed, right click on the feed shown in the above image, select **Edit feed** and click **Unsubscribe** in the dialog box that appears.

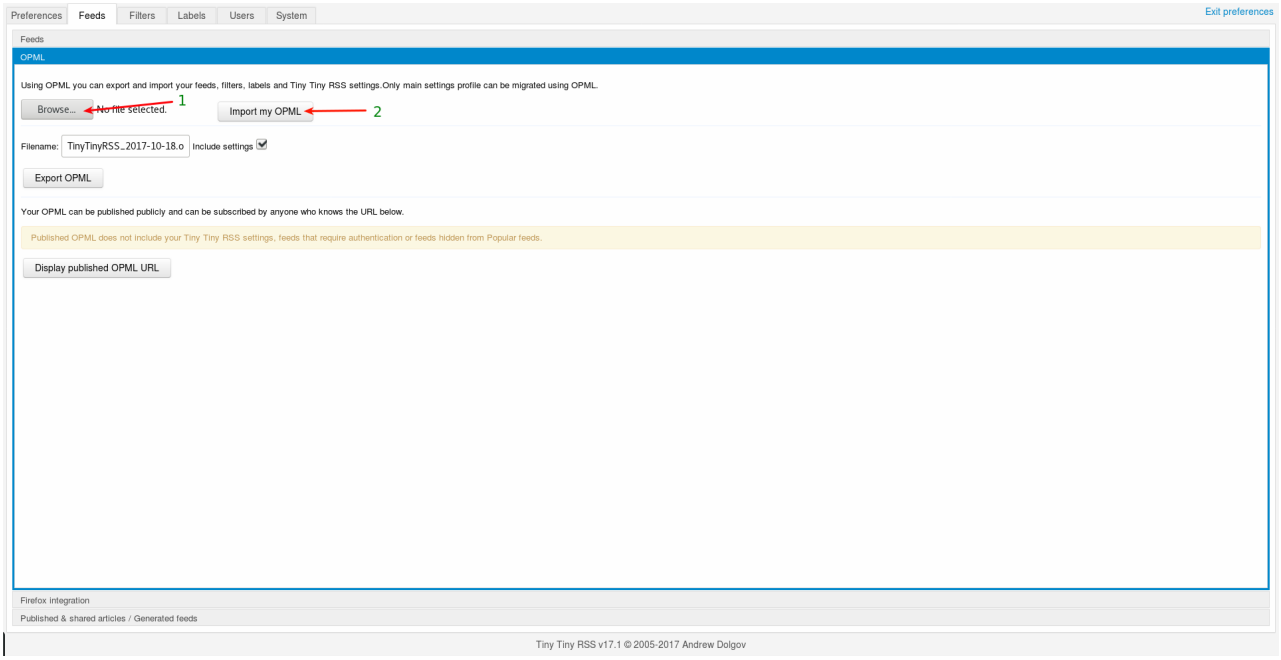


5.17.1.2 Importing your feeds from another feed reader

In your existing feed reader, find an option to *Export* your feeds to a file. Prefer the OPML file format if you have to choose between multiple formats. Let's say your exported feeds file is called Subscriptions.opml

Click on the *Actions* menu at the top left corner and select *Preferences*. You will be taken to another page.

Select the second tab called *Feeds* in the top header. Feeds has several sections. The second one is called *OPML*. Select it.



To import your Subscriptions.opml file into TT-RSS,

1. Click *Browse* and select the file from your file system

2. Click *Import my OPML*

After importing, you'll be taken to the **Feeds** section that's above the OPML section in the page. You can see that the feeds from your earlier feed reader are now imported into Tiny Tiny RSS. You can now start using Tiny Tiny RSS as your primary feed reader.

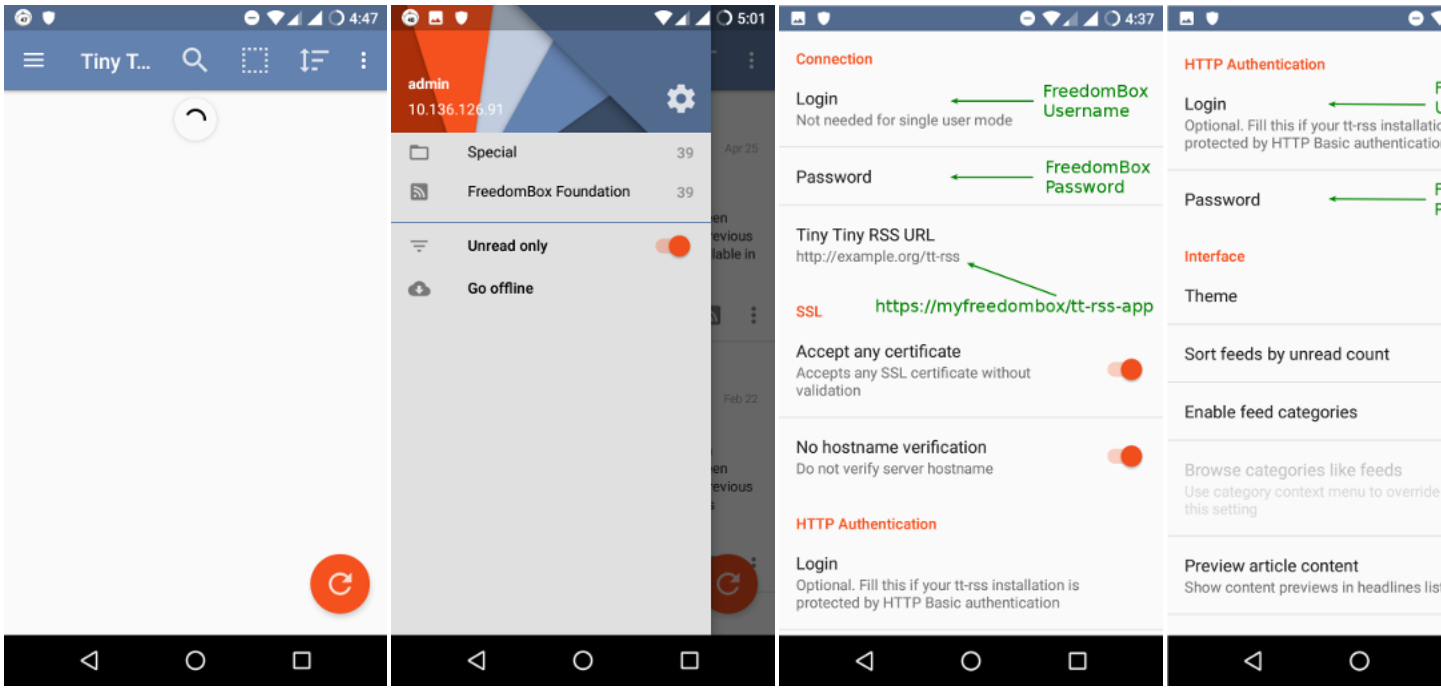
In the next section, we will discuss setting up the mobile app, which can let you read your feeds on the go.

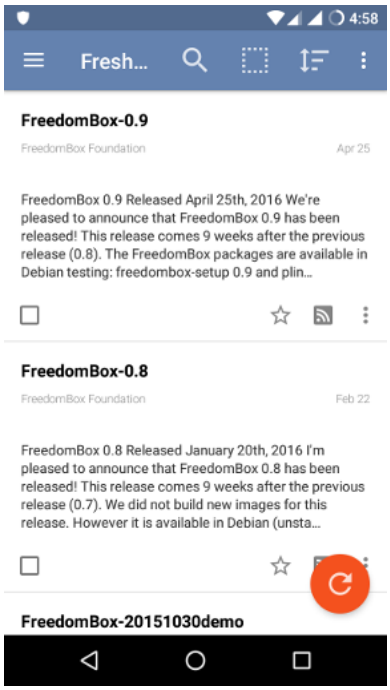
5.17.2 Using the Mobile App

The official Android app from the Tiny Tiny RSS project works with FreedomBox's Tiny Tiny RSS Server. The older TTRSS-Reader application is known **not** to work.

The official Android app is unfortunately only available on the Google Play Store and not on F-Droid. You can still obtain the source code and build the apk file yourself.

To configure, first install the application, then in the setting page, set URL as *https://<your.freedombox.address>/tt-rss-app/*. Set your user name and password in the Login details as well as HTTP Authentication details. If your FreedomBox does not have a valid HTTPS certificate, then in settings request allowing any SSL certificate and any host.





5.18 Simple Git Hosting (GitWeb)

Git is a distributed version-control system for tracking changes in source code during software development. GitWeb provides a web interface to Git repositories. You can browse history and content of source code, use search to find relevant commits and code. You can also clone repositories and upload code changes with a command-line Git client or with multiple available graphical clients. And you can share your code with people around the world.

To learn more on how to use Git visit Git tutorial.

Available since version: 19.19

5.18.1 Managing the repositories

After installation of GitWeb, a new repository can be created. It can be marked as *private* to limit access.

5.18.2 Access

GitWeb can be accessed after installation e.g. by the web client through `https://<my_freedombox_name>/gitweb`

5.18.3 HTTP basic auth

GitWeb on FreedomBox currently supports HTTP remotes only. To avoid having to enter the password each time you pull/push to the repository, you can edit your remote to include the credentials.

Example: `https://username:password@my.freedombox.rocks/gitweb/myrepo`

Your username and password will be encrypted. Someone monitoring the network traffic will notice the domain name only.

Note: If using this method, your password will be stored in plain text in the local repository's .git/config file. For this reason, you should create a FreedomBox user who has only access to the gitweb and never use an admin account.

5.18.4 Mirroring

Though your repositories are primarily hosted on your own FreedomBox, you can configure a repository on another Git hosting system like GitLab as a mirror.

5.19 SIP Server (repro)

Caution
App removed
repro has been removed from Debian 10 (Buster), and therefore is no longer available in FreedomBox.

5.20 SOCKS5 proxy (Shadowsocks)

5.20.1 What is Shadowsocks?

Shadowsocks is a lightweight and secure SOCKS5 proxy, designed to protect your Internet traffic. It can be used to bypass Internet filtering and censorship. Your FreedomBox can run a Shadowsocks client which can connect to a Shadowsocks server. It will also run a SOCKS5 proxy. Local devices can connect to this proxy, and their data will be encrypted and proxied through the Shadowsocks server.

Note: Shadowsocks is available in FreedomBox starting with Plinth version 0.18.

5.20.2 Using the Shadowsocks client?

The current implementation of Shadowsocks in FreedomBox only supports configuring FreedomBox as a Shadowsocks client. The current use case for Shadowsocks is as follows:

- Shadowsocks client (FreedomBox) is in a region where some parts of the Internet are blocked or censored.
- Shadowsocks server is in a different region, which doesn't have these blocks.
- The FreedomBox provides SOCKS proxy service on the local network for other devices to make use of its Shadowsocks connection.

At a future date it will be possible to configure FreedomBox as Shadowsocks server.

5.20.3 Configuring your FreedomBox for the Shadowsocks client

To enable Shadowsocks, first navigate to the Socks5 Proxy (Shadowsocks) page and install it.

Server: the Shadowsocks server is not the FreedomBox IP or URL; rather, it will be another server or VPS that has been configured as a Shadowsocks server. There are also some public Shadowsocks servers listed on the web, but be aware that whoever operates the server can see where requests are going, and any non-encrypted data will be visible to them.

To use Shadowsocks after setup, set the SOCKS5 proxy URL in your device, browser or application to `http://freedombox_address:1080/`

5.21 Virtual Private Network (OpenVPN)

5.21.1 What is OpenVPN?

OpenVPN provides to your FreedomBox a virtual private network service. You can use this software for remote access, site-to-site VPNs and Wi-Fi security. OpenVPN includes support for dynamic IP addresses and NAT.

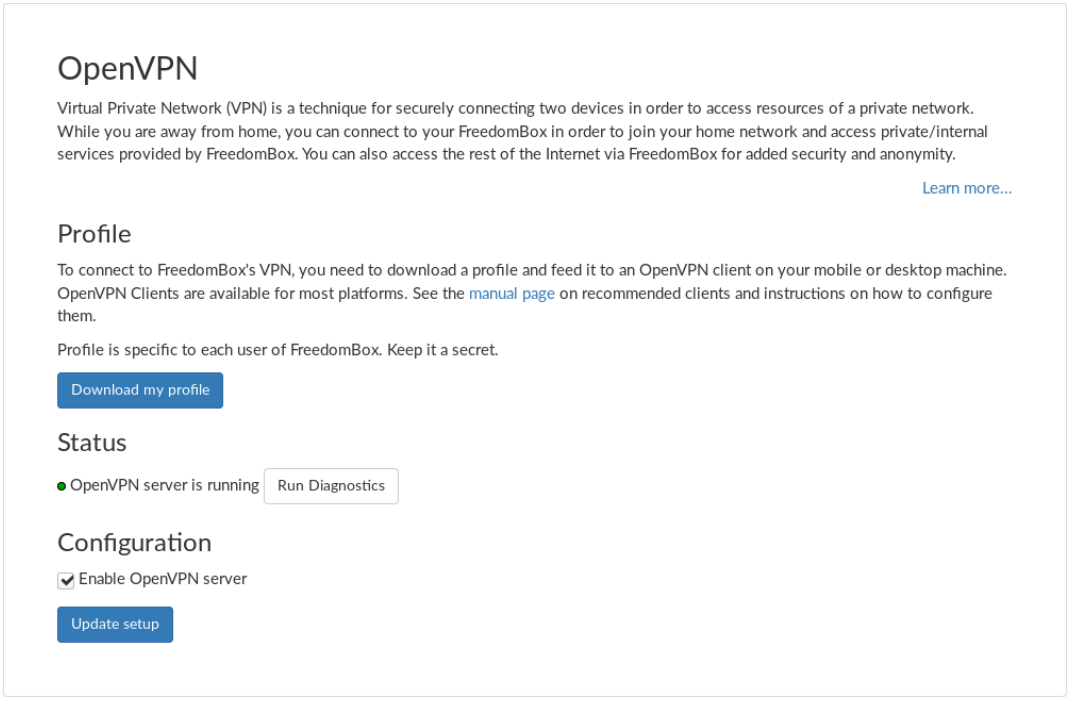
5.21.2 Port Forwarding

If your FreedomBox is behind a router, you will need to set up port forwarding on your router. You should forward the following ports for OpenVPN:

- UDP 1194

5.21.3 Setting up

1. In Plinth apps menu, select *Virtual Private Network (OpenVPN)* and click Install.
2. After the module is installed, there is an additional setup step that may take a long time to complete. Click "Start setup" to begin.



3. Wait for the setup to finish. This could take a while.
4. Once the setup of the OpenVPN server is complete, you can download your profile. This will download a file called <USER>.ovpn, where <USER> is the name of a FreedomBox user. Each FreedomBox user will be able to download a different profile. Users who are not administrators can download the profile from home page after login.
5. The ovpn file contains all the information a vpn client needs to connect to the server.
6. The downloaded profile contains the domain name of the FreedomBox that the client should connect to. This is picked up from the domain configured in 'Config' section of 'System' page. In case your domain is not configured properly, you may need to change this value after downloading the profile. If your OpenVPN client allows it, you can do this after importing the OpenVPN profile. Otherwise, you can edit the .ovpn profile file in a text editor and change the 'remote' line to contain the WAN IP address or hostname of your FreedomBox as follows.

```
client
remote mybox.sds-ip.de 1194
proto udp
```

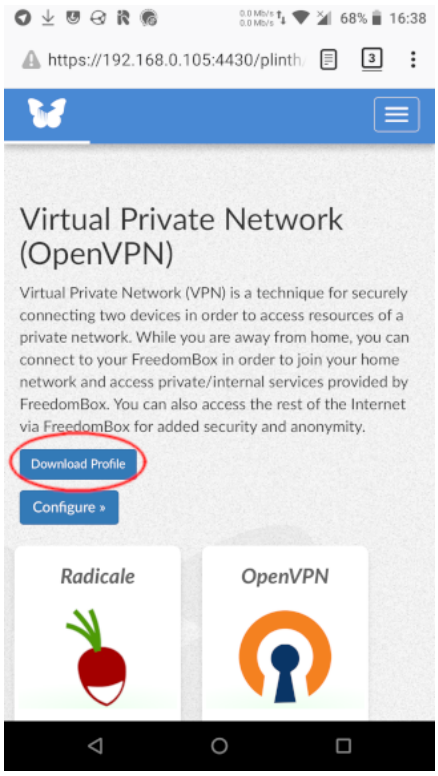
5.21.4 Browsing Internet after connecting to VPN

After connecting to the VPN, the client device will be able to browse the Internet without any further configuration. However, a pre-condition for this to work is that you need to have at least one Internet connected network interface which is part of the 'External' firewall zone. Use the networks configuration page to edit the firewall zone for the device's network interfaces.

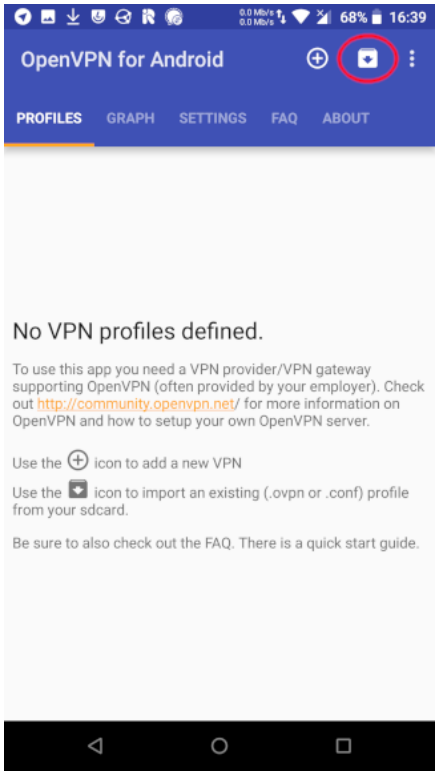
5.21.5 Usage

5.21.5.1 On Android/LineageOS

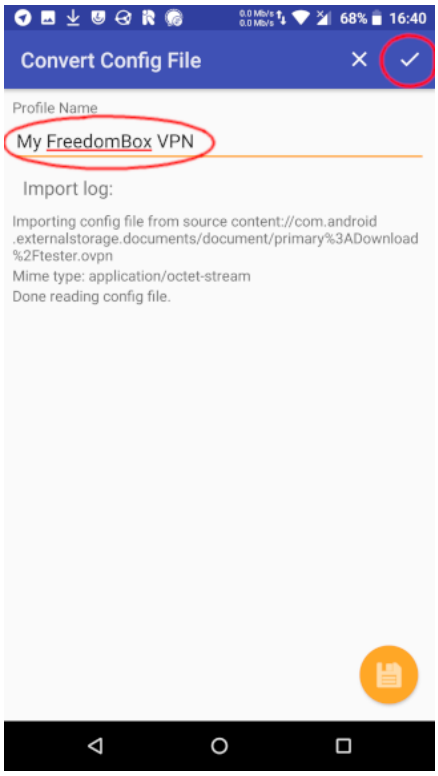
1. Visit FreedomBox home page. Login with your user account. From home page, download the OpenVPN profile. The file will be named *username*.ovpn.



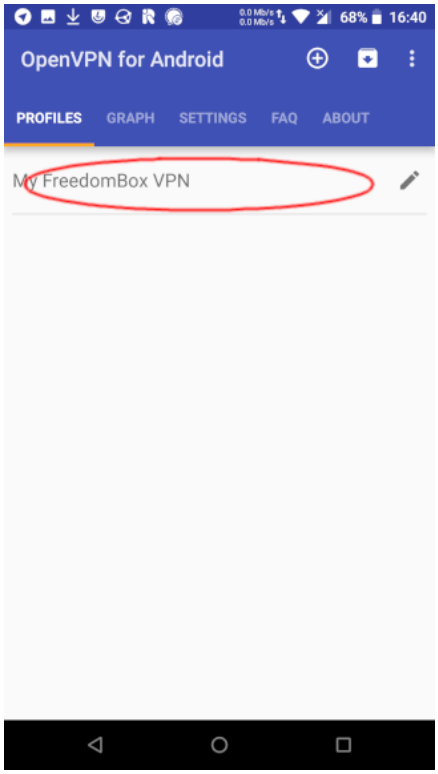
2. Download an OpenVPN client such as *OpenVPN for Android*. F-Droid repository is recommended. In the app, select import profile.

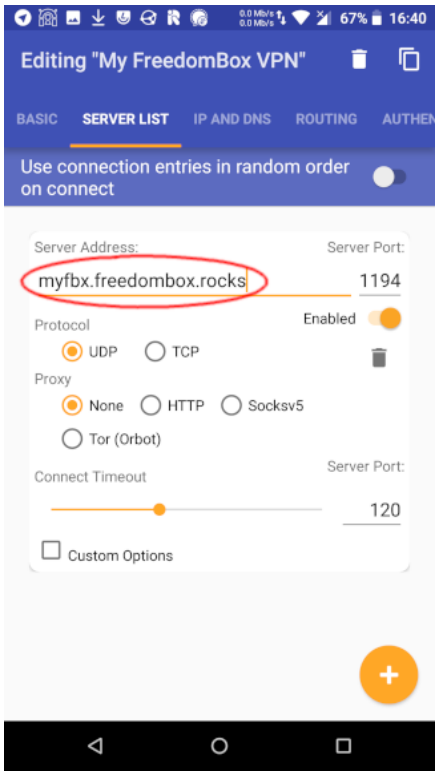


3. In the select profile dialog, choose the *username*.opvn file you have just downloaded. Provide a name for the connection and save the profile.

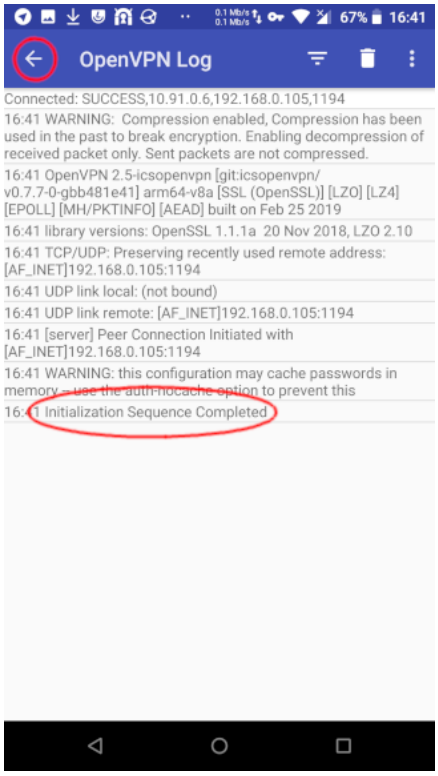


4. Newly created profile will show up. If necessary, edit the profile and set the domain name of your FreedomBox as the server address.

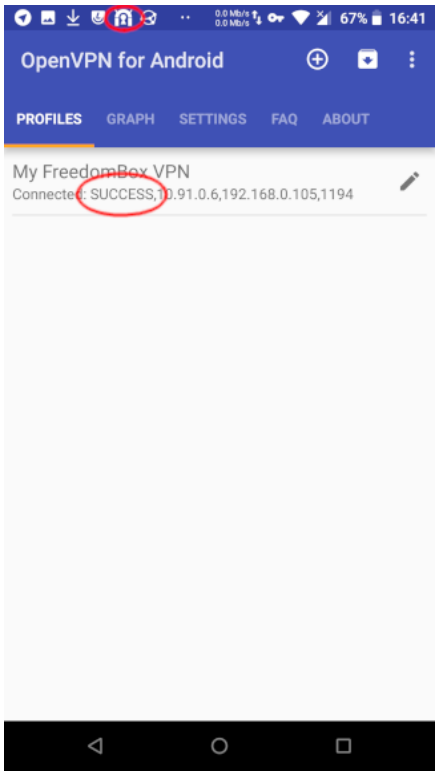




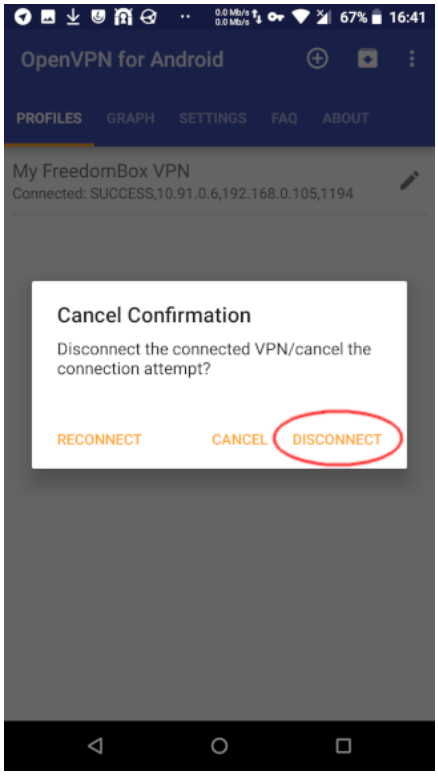
5. Connect by tapping on the profile.



-



6. When done, disconnect by tapping on the profile.



5.21.5.2 On Debian

Install an OpenVPN client for your system

```
$ sudo apt install openvpn
```

Open the ovpn file with the OpenVPN client.

```
$ sudo openvpn --config /path/to/<USER>.ovpn
```

If you use Network Manager, you can create a new connection by importing the file:

```
$ sudo apt install network-manager-openvpn-gnome
$ sudo nmcli connection import type openvpn file /path/to/<USER>.ovpn
```

5.21.6 Checking if you are connected

5.21.6.1 On Debian

1. Try to ping the FreedomBox or other devices on the local network.
2. Running the command `ip addr` should show a `tun0` connection.
3. The command `traceroute freedombox.org` should show you the ip address of the VPN server as the first hop.

5.21.7 External Links

https://community.openvpn.net/openvpn

5.22 Voice Chat (Mumble)

5.22.1 What is Mumble?

Mumble is a voice chat software. Primarily intended for use while gaming, it is suitable for simple talking with high audio quality, noise suppression, encrypted communication, public/private-key authentication by default, and "wizards" to configure your microphone for instance. A user can be marked as a "priority speaker" within a channel.

5.22.2 Using Mumble

FreedomBox includes the Mumble server. Clients are available for desktop and mobile platforms. Users can download one of these clients and connect to the server.

5.22.3 Port Forwarding

If your FreedomBox is behind a router, you will need to set up port forwarding on your router. You should forward the following ports for Mumble:

- TCP 64738
- UDP 64738

5.22.4 Managing Permissions

A super user in Mumble has the ability to create administrator accounts who can in turn manage groups and channel permissions. This can be done after logging in with the username "SuperUser" using the super user password. See Mumble Guide for information on how to do this.. FreedomBox currently does not offer a UI to get or set the super user password for Mumble. A super user password is automatically generated during Mumble setup. To get the password, login to the terminal as admin user using Cockpit , Secure Shell or the console. Then, to read the super user password that was automatically generated during Mumble installation run the following command:

```
sudo grep SuperUser /var/log/mumble-server/mumble-server.log
```

You should see output such as:

```
<W>2019-11-06 02:47:41.313 1 => Password for 'SuperUser' set to 'noo8Dahwiesh'
```

Alternatively, you can set a new password as follows:

```
sudo su -
echo "newpassword" | su mumble-server -s /bin/sh -c "/usr/sbin/murmurd -ini /etc/mumble- ↩
    server.ini --readsupw"
```

5.23 Web Proxy (Privoxy)

A web proxy acts as a filter for incoming and outgoing web traffic. Thus, you can instruct any computer in your network to pass internet traffic through the proxy to remove unwanted ads and tracking mechanisms.

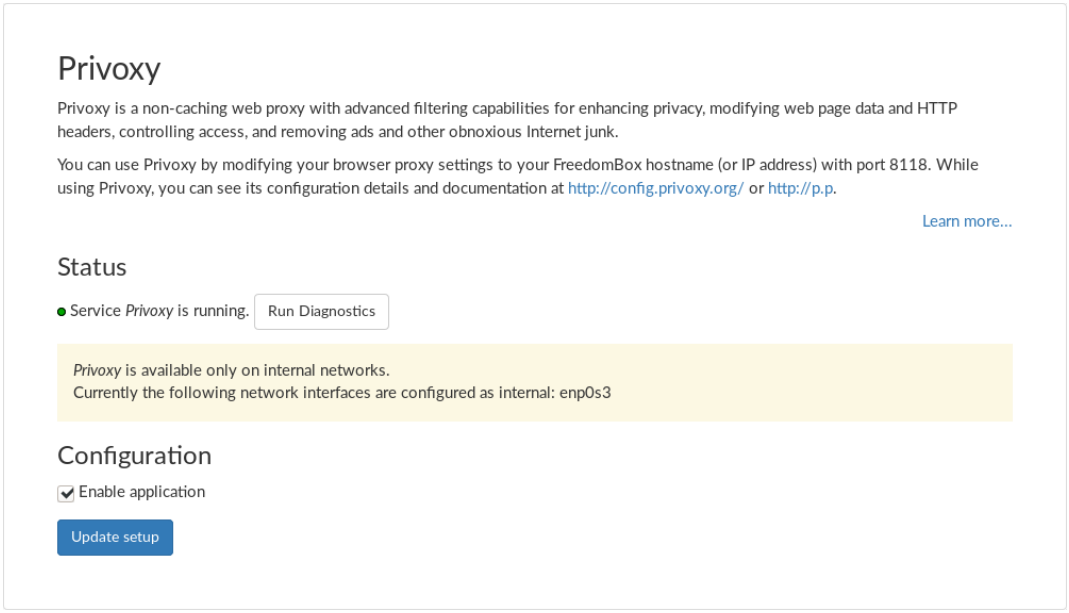
Privoxy is a software for security, privacy, and accurate control over the web. It provides a much more powerful web proxy (and anonymity on the web) than what your browser can offer. Privoxy "is a proxy that is primarily focused on privacy enhancement, ad and junk elimination and freeing the user from restrictions placed on his activities" (source: Privoxy FAQ).

5.23.1 Screencast

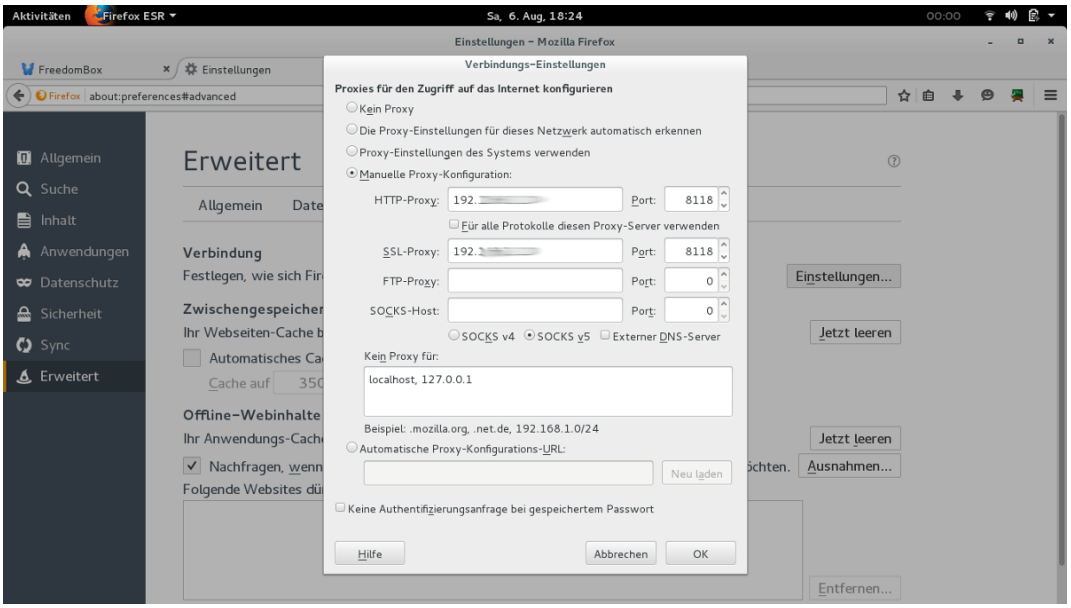
Watch the screencast on how to setup and use Privoxy in FreedomBox.

5.23.2 Setting up

1. In Plinth install *Web Proxy (Privoxy)*



2. Adapt your browser proxy settings to your FreedomBox hostname (or IP address) with port 8118. Please note that Privoxy can only proxy HTTP and HTTPS traffic. It will not work with FTP or other protocols.



3. Go to page `http://config.privoxy.org/` or `http://p.p`. If Privoxy is installed properly, you will be able to configure it in detail; if not you will see an error message.

4. If you are using a laptop that occasionally has to connect through other routers than yours with the FreedomBox and Privoxy, you may want to install a proxy switch add-on that allows you to easily turn the proxy on or off.

5.23.3 Advanced Users

1. The default installation should provide a reasonable starting point for most. There will undoubtedly be occasions where you will want to adjust the configuration, that can be dealt with as the need arises.

2. While using Privoxy, you can see its configuration details and documentation at `http://config.privoxy.org/` or `http://p.p`.

3. To enable changing these configurations, you first have to change the value of `enable-edit-actions` in `/etc/privoxy/c` to `1`. Before doing so, read carefully the manual, especially:

   - *Access to the editor can not be controlled separately by "ACLs" or HTTP authentication, so that everybody who can access Privoxy can modify its configuration for all users. This option is not recommended for environments with untrusted users. Note that malicious client side code (e.g Java) is also capable of using the actions editor and you shouldn't enable this options unless you understand the consequences and are sure your browser is configured correctly.*

4. Now you find an `EDIT` button on the configuration screen in `http://config.privoxy.org/`.

5. The Quickstart is a good starting point to read on how to define own blocking and filtering rules.

5.24 Web Search (Searx)

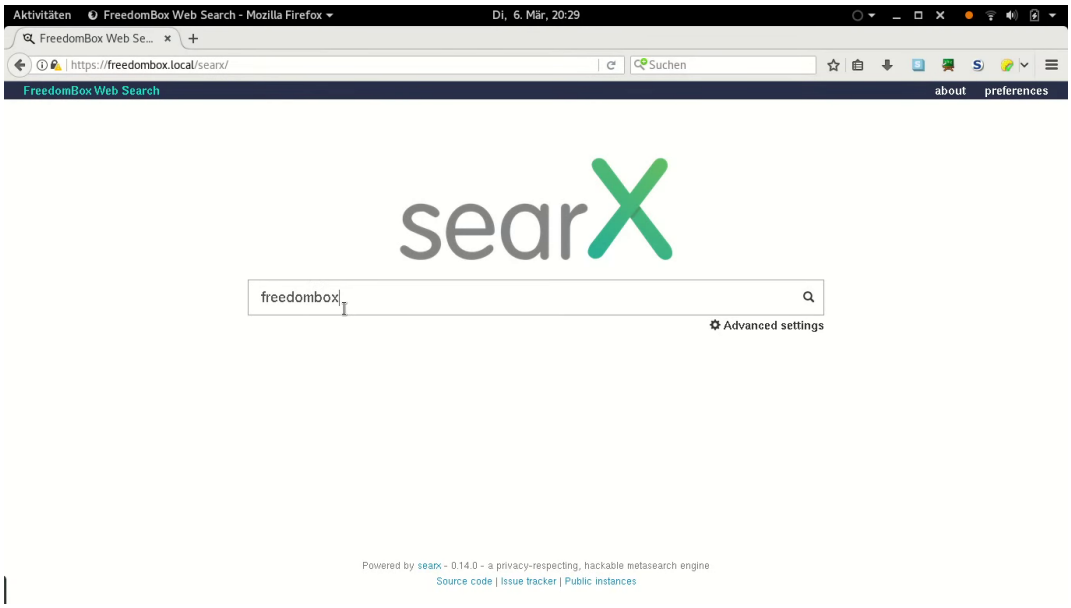
5.24.1 About Searx

Searx is a metasearch engine. A metasearch engine aggregates the results from various search engines and presents them in a unified interface.

Read more about Searx on their official website.

Available since: version 0.24.0

5.24.2 Screenshot



5.24.3 Screencast

Searx installation and first steps (14 MB)

5.24.4 Why use Searx?

5.24.4.1 Personalization and Filter Bubbles

Search engines have the ability to profile users and serve results most relevant to them, putting people into filter bubbles, thus distorting people's view of the world. Search engines have a financial incentive to serve interesting advertisements to their users, increasing their chances of clicking on the advertisements.

A metasearch engine is a possible solution to this problem, as it aggregates results from multiple search engines thus bypassing personalization attempts by search engines.

Searx avoids storing cookies from search engines as a means of preventing tracking and profiling by search engines.

5.24.4.2 Advertisement filtering

Searx filters out advertisements from the search results before serving the results, thus increasing relevance the of your search results and saving you from distractions.

5.24.4.3 Privacy

Searx uses HTTP POST instead of GET by default to send your search queries to the search engines, so that anyone snooping your traffic wouldn't be able to read your queries. The search queries wouldn't stored in browser history either.

Note: Searx used from Chrome browser's omnibar would make GET requests instead of POST.

5.24.5 Searx on FreedomBox

- Searx on FreedomBox uses Single Sign On. This means that you should be logged in into your FreedomBox in the browser that you're using Searx.

- SearX is easily accessible via Tor.
- Searx can be added as a search engine to the Firefox browser's search bar. See Firefox Help on this topic. Once Searx is added, you can also set it as your default search engine.
- Searx also offers search results in csv, json and rss formats, which can be used with scripts to automate some tasks.

5.25 Wiki (MediaWiki)

5.25.1 About MediaWiki

MediaWiki is the software that powers the Wikimedia suite of wikis.

Read more about MediaWiki on Wikipedia

Available since: version 0.20.0

5.25.2 MediaWiki on FreedomBox

MediaWiki on FreedomBox is configured to be publicly readable and privately editable. Only logged in users can make edits to the wiki. This configuration prevents spam and vandalism on the wiki.

5.25.2.1 User management

Users can be created by the MediaWiki administrator (user "admin") only. The "admin" user can also be used to reset passwords of MediaWiki users. The administrator password, if forgotten can be reset anytime from the MediaWiki app page in web interface.

5.25.2.2 Use cases

MediaWiki is quite versatile and can be put to many creative uses. It also comes with a lot of plugins and themes and is highly customizable.

5.25.2.2.1 Personal Knowledge Repository

- MediaWiki on FreedomBox can be your own personal knowledge repository. Since MediaWiki has good multimedia support, you can write notes, store images, create checklists, store references and bookmarks etc. in an organized manner. You can store the knowledge of a lifetime in your MediaWiki instance.

5.25.2.2.2 Community Wiki

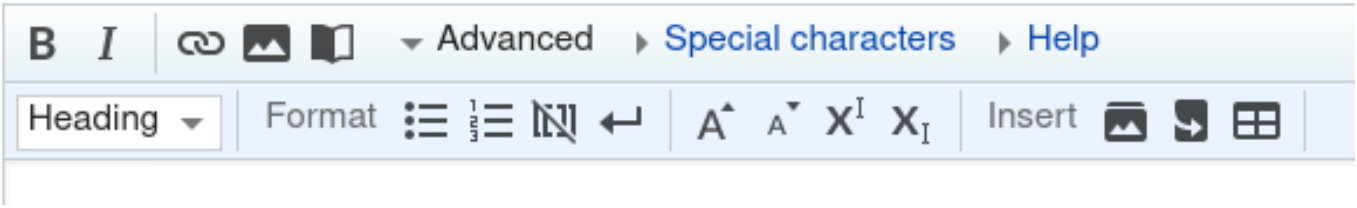
- A community of users can use MediaWiki as their common repository of knowledge and reference material. It can used as a college notice board, documentation server for a small company, common notebook for study groups or as a fan wiki like wikia.

5.25.2.2.3 Personal Wiki-based Website

- Several websites on the internet are simply MediaWiki instances. MediaWiki on FreedomBox is read-only to visitors. Hence, it can be adapted to serve as your personal website and/or blog. MediaWiki content is easy to export and can be later moved to use another blog engine.

5.25.2.3 Editing Wiki Content

The MediaWiki installation on FreedomBox ships with a basic editor with a toolbar for common options like Bold, Italics etc. Click on the Advanced section for more options like Headings, bullet lists etc.



5.25.2.3.1 Visual Editor

- MediaWiki's new Visual Editor gives a WYSIWYG user interface to creating wiki pages. This is still a Beta feature and is not provided by default with MediaWiki. A workaround is to use write your content using the Visual Editor in Wikipedia's Sandbox, switching to source editing mode and copying the content into your wiki.

5.25.2.3.2 Other Formats

- You don't have to necessarily learn the MediaWiki formatting language. You can write in your favorite format (Markdown, Org-mode, LaTeX etc.) and convert it to the MediaWiki format using Pandoc.

5.25.2.3.3 Image Uploads

- Image uploads have been enabled since FreedomBox version 0.36.0. You can also directly use images from Wikimedia Commons using a feature called Instant Commons.

5.25.2.4 Customization

5.25.2.4.1 Skins

MediaWiki's default skin is usually Vector. The default skin set by FreedomBox is Timeless.

Vector is a skin best-suited for viewing on desktop browsers. It is not suitable for mobile screen sizes. Wikimedia sites host a separate mobile site. It is not worth hosting a separate mobile site for small MediaWiki installations like those on FreedomBox. Using a mobile-friendly skin like Timeless is a cheaper way of solving the problem.

Administrators can choose a default skin from the app configuration. Users of the site also have the choice of viewing it with a different skin.

5.26 Wiki and Blog (Ikiwiki)

5.26.1 What is Ikiwiki?

Ikiwiki converts wiki pages into HTML pages suitable for publishing on a website. It provides particularly blogging, podcasting, calendars and a large selection of plugins.

5.26.2 Quick Start

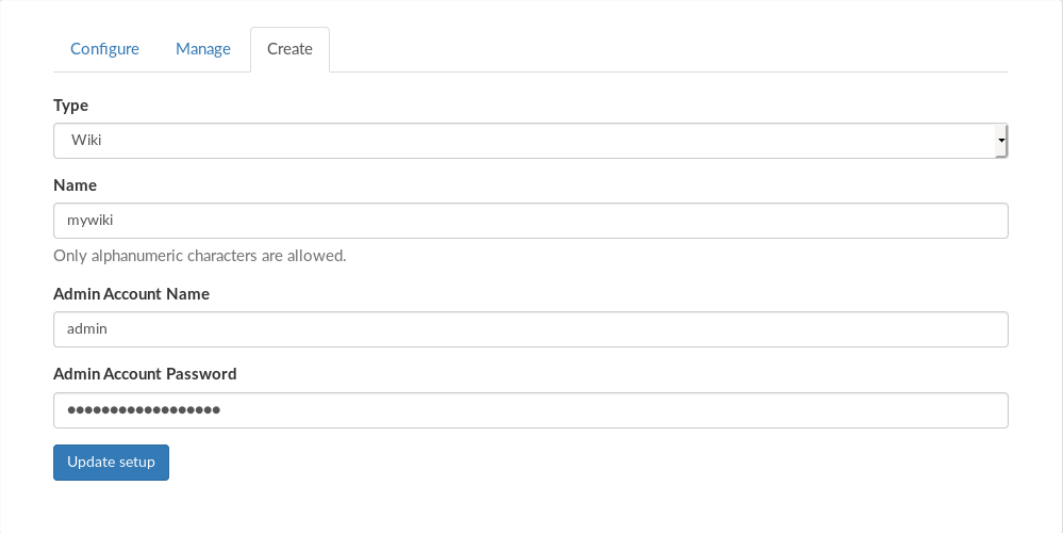
After the app installation on your box administration interface:

- Go to "Create" section and create a wiki or a blog
- Go back to "Configure" section and click on /ikiwiki link
- Click on your new wiki or blog name under "Parent directory"
- Enjoy your new publication page.

5.26.3 Creating a wiki or blog

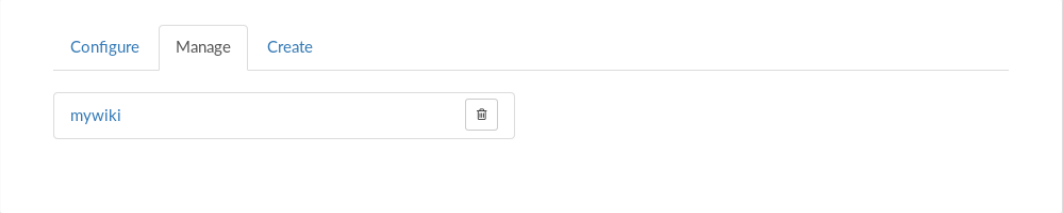
You can create a wiki or blog to be hosted on your FreedomBox through the Wiki & Blog (Ikiwiki) page in Plinth. The first time you visit this page, it will ask to install packages required by Ikiwiki.

After the package install has completed, select the Create tab. You can select the type to be Wiki or Blog. Also type in a name for the wiki or blog, and the username and password for the wiki's/blog's admin account. Then click Update setup and you will see the wiki/blog added to your list. Note that each wiki/blog has its own admin account.



5.26.4 Accessing your wiki or blog

From the Wiki & Blog (Ikiwiki) page, select the Manage tab and you will see a list of your wikis and blogs. Click a name to navigate to that wiki or blog.



From here, if you click Edit or Preferences, you will be taken to a login page. To log in with the admin account that you created before, select the Other tab, enter the username and password, and click Login.

5.26.5 User login through SSO

Besides the wiki/blog admin, other FreedomBox users can be given access to login and edit wikis and blogs. However, they will not have all the same permissions as the wiki admin. They can add or edit pages, but cannot change the wiki's configuration.

To add a wiki user, go to the Users and Groups page in Plinth (under System configuration, the gear icon at the top right corner of the page). Create or modify a user, and add them to the wiki group. (Users in the admin group will also have wiki access.)

To login as a FreedomBox user, go to the wiki/blog's login page and select the Other tab. Then click the "Login with HTTP auth" button. The browser will show a popup dialog where you can enter the username and password of the FreedomBox user.

5.26.6 Adding FreedomBox users as wiki admins

1. Login to the wiki, using the admin account that was specified when the wiki was created.

2. Click "Preferences", then "Setup".
3. Under "main", in the "users who are wiki admins", add the name of a user on the FreedomBox.
4. (Optional) Under "auth plugin: passwordauth", uncheck the "enable passwordauth?" option. (Note: This will disable the old admin account login. Only SSO login using HTTP auth will be possible.)
5. Click "Save Setup".
6. Click "Preferences", then "Logout".
7. Login as the new admin user using "Login with HTTP auth".

5.27 Anonymity Network (I2P)

5.27.1 About I2P

The Invisible Internet Project is an anonymous network layer intended to protect communication from censorship and surveillance. I2P provides anonymity by sending encrypted traffic through a volunteer-run network distributed around the world.

Find more information about I2P on their project homepage.

5.27.2 Services Offered

The following services are offered via I2P in FreedomBox by default. Additional services may be available when enabled from I2P router console that can be launched from FreedomBox web interface.

- **Anonymous Internet browsing**: I2P can be used to browse Internet anonymously. For this, configure your browser (preferable a Tor Browser) to connect to I2P proxy. This can be done by setting HTTP proxy and HTTPS proxy to *freedombox.local* (or your FreedomBox's local IP address) and ports to *4444* and *4445* respectively. This service is available only when you are reaching FreedomBox using local network (networks in internal zone) and not available when connecting to FreedomBox from the Internet. One exception to this is when you connect to FreedomBox's VPN service from Internet you can still use this service.
- **Reaching eepsites**: I2P network can host websites that can remain anonymous. These are called eepsites and end with .i2p in their domain name. For example, `http://i2p-projekt.i2p/` is the website for I2P project in the I2P network. eepsites are not reachable using a regular browser via regular Internet connection. To browse eepsites, your browser needs to be configured to use HTTP, HTTPS proxies as described above. This service is available only when you are reaching FreedomBox using local network (networks in internal zone) and not available when connecting to FreedomBox from the Internet. One exception to this is when you connect to FreedomBox's VPN service from Internet you can still use this service.
- **Anonymous torrent downloads**: I2PSnark, an application for anonymously downloading and sharing files over the BitTorrent network is available in I2P and enabled by default in FreedomBox. This application is controlled via a web interface that can be launched from 'Anonymous torrents' section of I2P app in FreedomBox web interface or from the I2P router console interface. Only logged-in users belonging to 'Manage I2P application' group can use this service.
- **IRC network**: I2P network contains an IRC network called Irc2P. This network hosts the I2P project's official IRC channel among other channels. This service is enabled by default in FreedomBox. To use it, open your favourite IRC client. Then configure it to connect to host *freedombox.local* (or your FreedomBox's local IP address) with port number *6668*. This service is available only when you are reaching FreedomBox using local network (networks in internal zone) and not available when connecting to FreedomBox from the Internet. One exception to this is when you connect to FreedomBox's VPN service from Internet you can still use this service.
- **I2P router console**: This is the central management interface for I2P. It shows the current status of I2P, bandwidth statistics and allows modifying various configuration settings. You can tune your participation in the I2P network and use/edit a list of your favourite I2P sites (eepsites). Only logged-in users belonging to 'Manage I2P application' group can use this service.

6 System

6.1 Backups

FreedomBox includes the ability to backup and restore data, preferences, configuration and secrets from most of the applications. The Backups feature is built using Borg backup software. Borg is a deduplicating and compressing backup program. It is designed for efficient and secure backups. This backups feature can be used to selectively backup and restore data on an app-by-app basis. Backed up data can be stored on the FreedomBox machine itself or on a remote server. Any remote server providing SSH access can be used as a backup storage repository for FreedomBox backups. Data stored remotely may be encrypted and in such cases remote server cannot access your decrypted data.

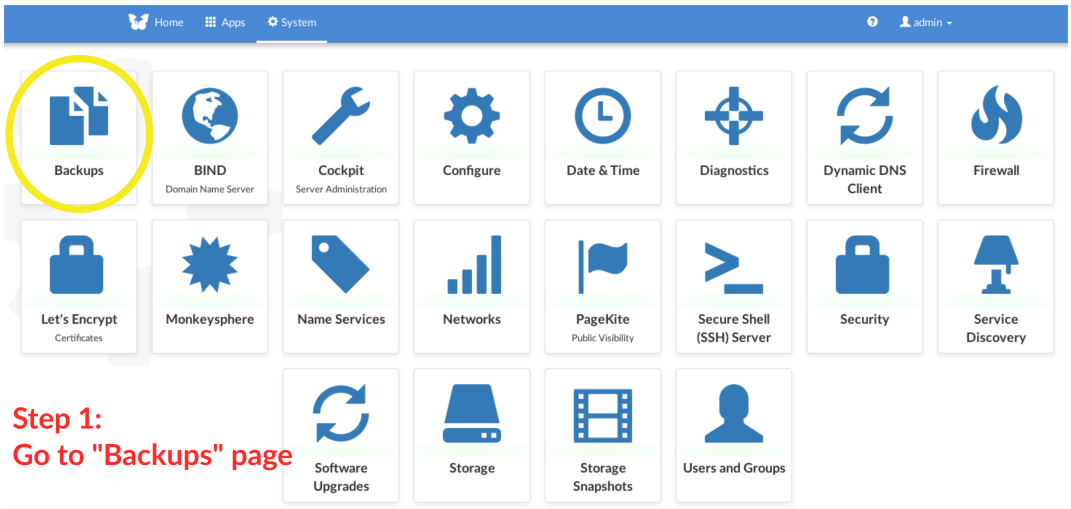
6.1.1 Status of Backups Feature

App/Feature	Support in Version	Notes
Avahi	-	no backup needed
Backups	-	no backup needed
Bind	0.41	
Cockpit	-	no backup needed
Coquelicot	0.40	includes uploaded files
Datetime	0.41	
Deluge	0.41	does not include downloaded/seeding files
Diagnostics	-	no backup needed
Dynamic DNS	0.39	
ejabberd	0.39	includes all data and configuration
Firewall	-	no backup needed
ikiwiki	0.39	includes all wikis/blogs and their content
infinoted	0.39	includes all data and keys
JSXC	-	no backup needed
Let's Encrypt	0.42	
Matrix Synapse	0.39	includes media and uploads
MediaWiki	0.39	includes wiki pages and uploaded files
Minetest	0.39	
MLDonkey	19.0	
Monkeysphere	0.42	
Mumble	0.40	
Names	-	no backup needed
Networks	No	No plans currently to implement backup
OpenVPN	0.48	includes all user and server keys
Pagekite	0.40	
Power	-	no backup needed
Privoxy	-	no backup needed
Quassel	0.40	includes users and logs
Radicale	0.39	includes calendar and cards data for all users
repro	0.39	includes all users, data and keys
Roundcube	-	no backup needed
SearX	-	no backup needed
Secure Shell (SSH) Server	0.41	includes host keys
Security	0.41	
Shadowsocks	0.40	only secrets
Sharing	0.40	does not include the data in the shared folders

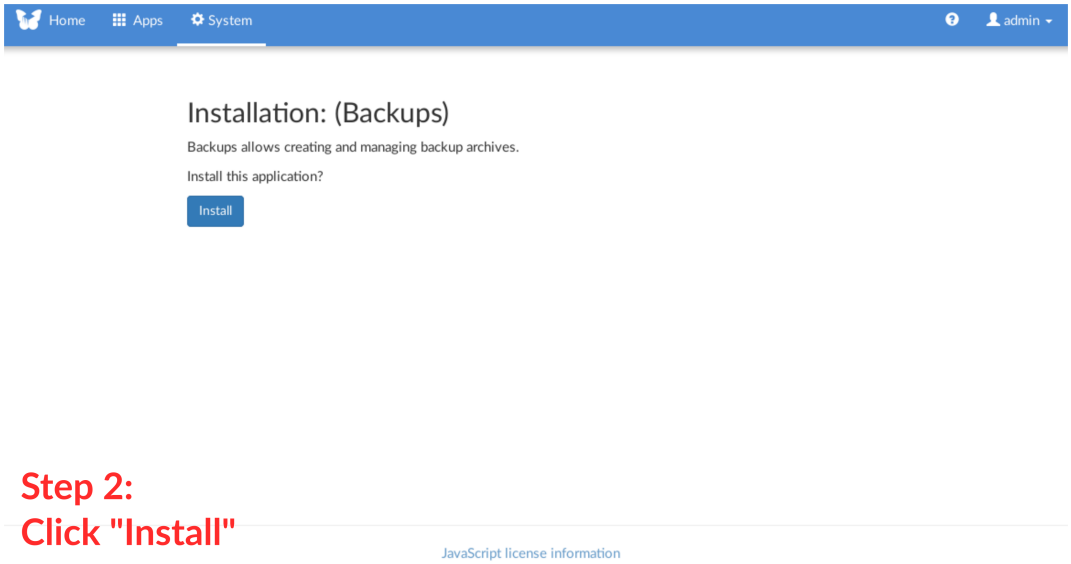
Snapshot	0.41	only configuration, does not include snapshot data
Storage	-	no backup needed
Syncthing	0.48	does not include data in the shared folders
Tahoe-LAFS	0.42	includes all data and configuration
Tiny Tiny RSS	19.2	includes database containing feeds, stories, etc.
Tor	0.42	includes configuration and secrets such as onion service keys
Transmission	0.40	does not include downloaded/seeding files
Upgrades	0.42	
Users	No	No plans currently to implement backup

6.1.2 How to install and use Backups

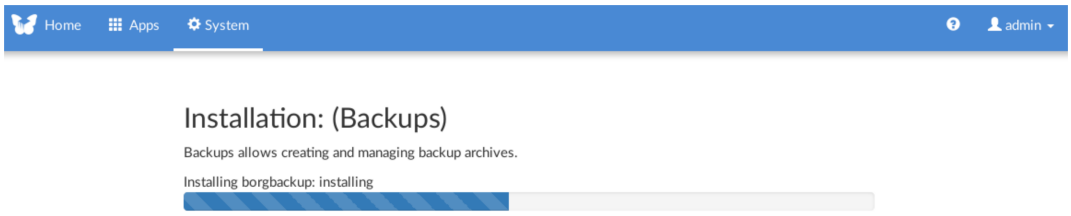
Step 1



Step 2



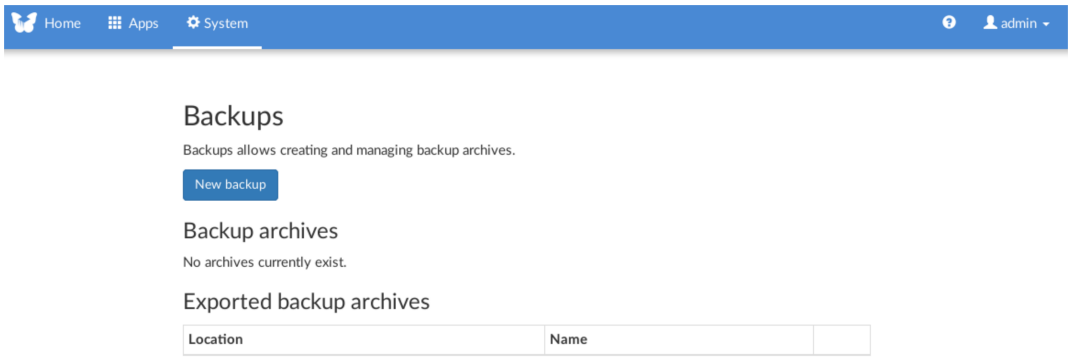
Step 3



Step 3:
Wait for Backups to install

JavaScript license information

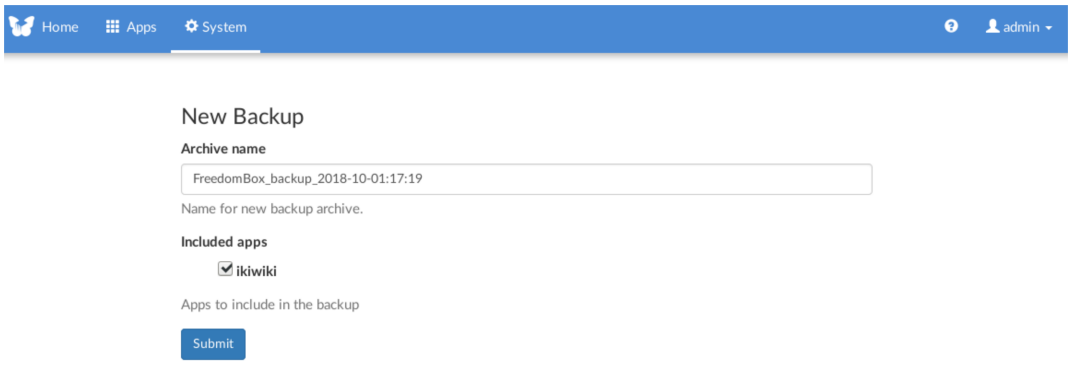
Step 4



Step 4:
Click "New backup" to create the backup archive

JavaScript license information

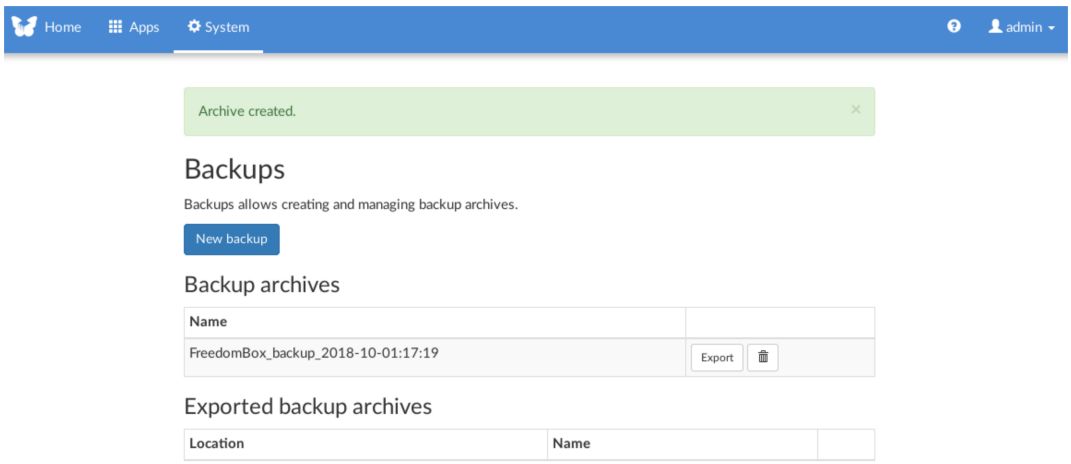
Step 5



Step 5:
Type a name for the archive, select apps to include, click submit

JavaScript license information

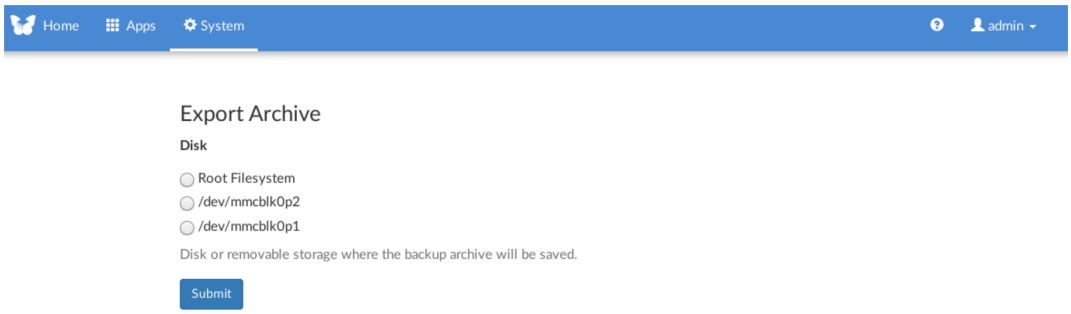
Step 6



Step 6:
Click "Export" to export the archive

JavaScript license information

Step 7



6.2 Configure

Configure has some general configuration options:

6.2.1 Hostname

- Hostname is the local name by which other devices on the local network can reach your FreedomBox. The default hostname is *freedombox*.

6.2.2 Domain Name

- Domain name is the global name by which other devices on the Internet can reach your FreedomBox. The value set here is used by the Chat Server (XMPP), Matrix Synapse, Certificates (Let's Encrypt), and Monkeysphere.

6.2.3 Webserver Home Page

- This is an advanced option that allows you to set something other than FreedomBox Service (Plinth) as the home page to be served on the domain name of the FreedomBox. For example, if your FreedomBox's domain name is `https://myfreedombox.rocks` and you set MediaWiki as the home page, visiting `https://myfreedombox.rocks` will take you to `https://myfreedombox.rocks/mediawiki/` instead of the usual `https://myfreedombox.rocks/plinth/`. You can set any web application, Ikiwiki wikis and blogs or Apache's default index.html page as the web server home page.

Caution
Once some other app is set as the home page, you can only navigate to the FreedomBox Service (Plinth) by typing `https://myfreedombox.rocks/plinth/` into the browser.
/freedombox can also be used as an alias to */plinth*

- *Tip:* Bookmark the URL of FreedomBox Service (Plinth) before setting the home page to some other app.

6.3 Cockpit (Server Administration)

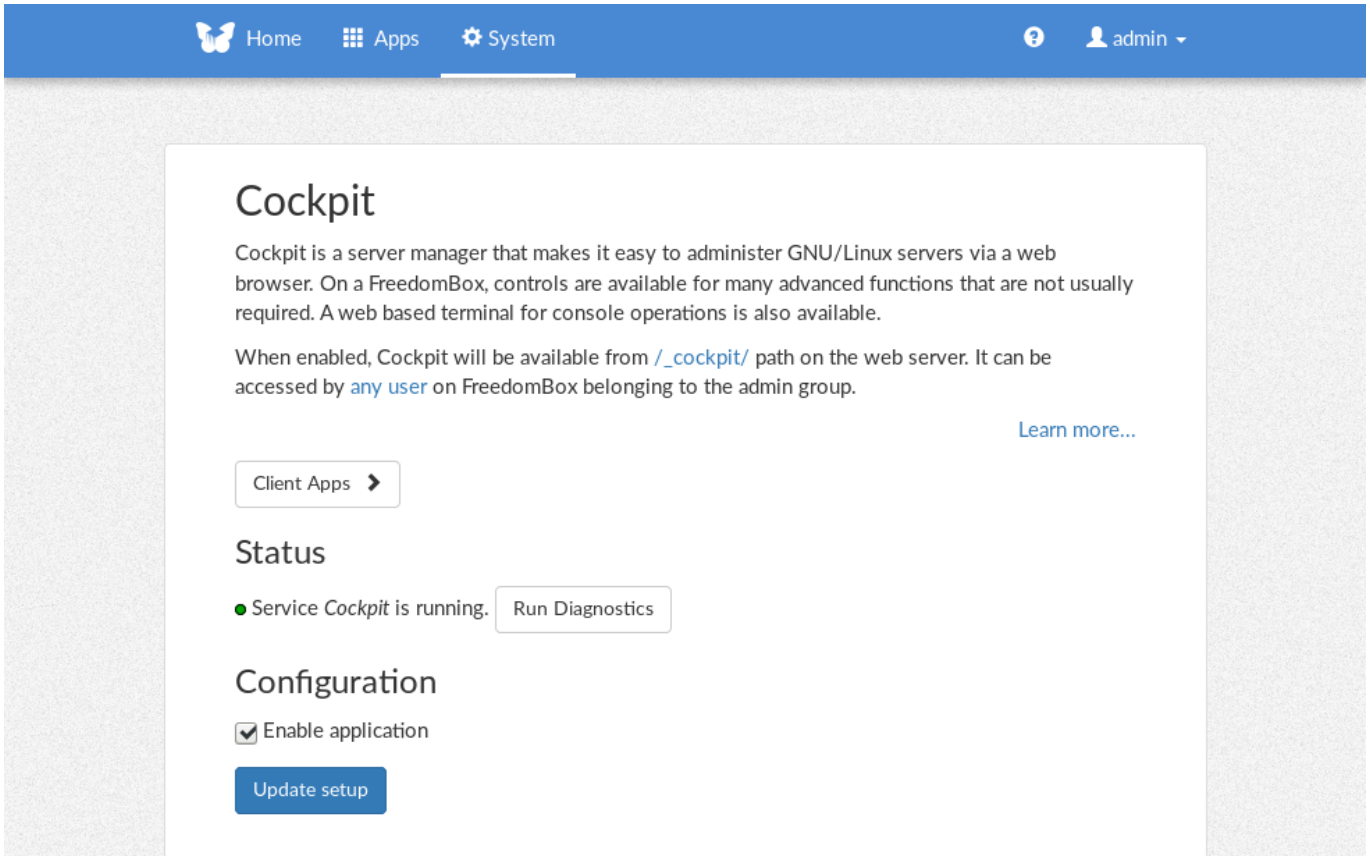
Cockpit is a server manager that makes it easy to administer GNU/Linux servers via a web browser. On a FreedomBox, controls are available for many advanced functions that are not usually required. A web based terminal for console operations is also available.

It can be accessed by any user on your FreedomBox belonging to the admin group. Cockpit is only usable when you have proper domain name setup for your FreedomBox and you use that domain name to access Cockpit. See the Troubleshooting section for more information.

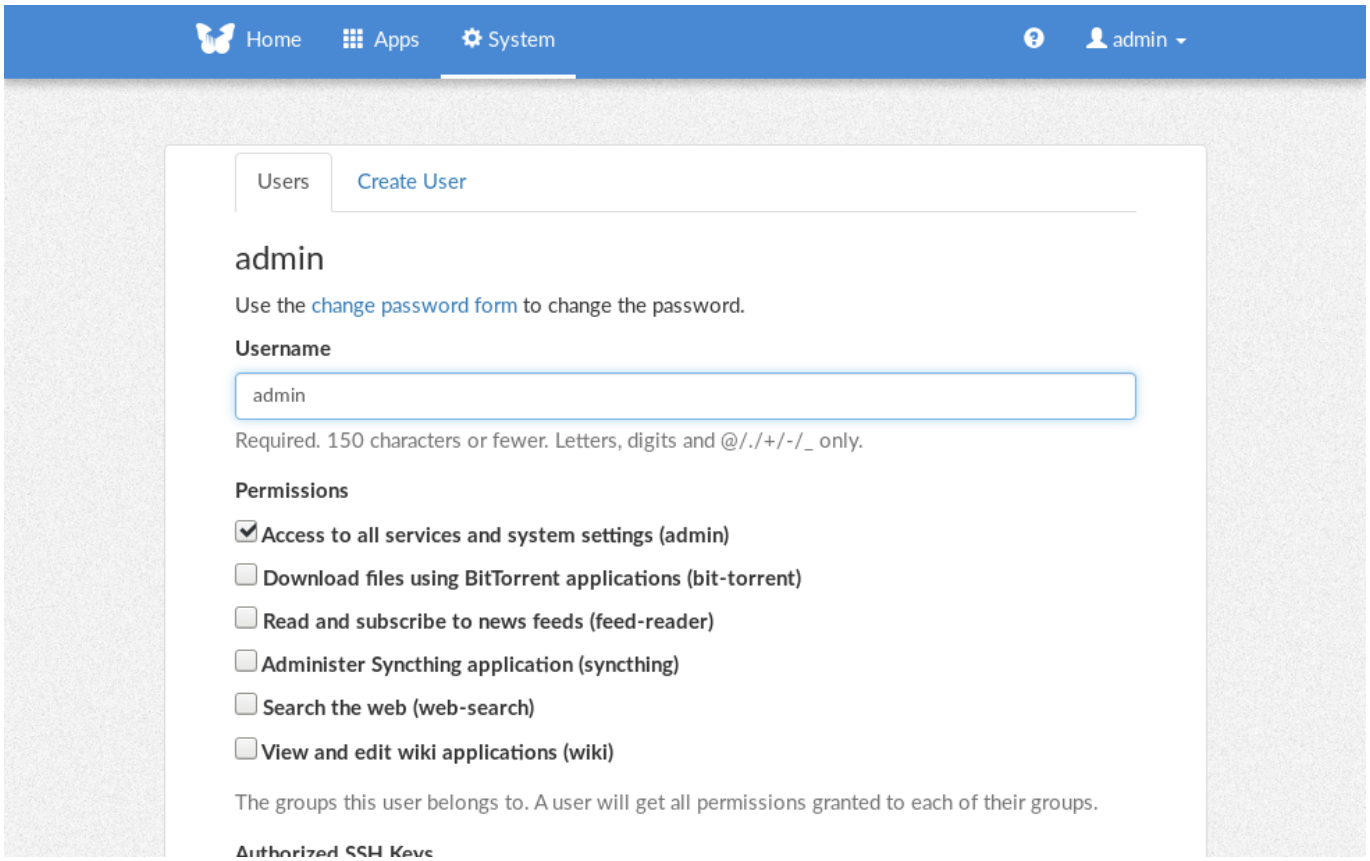
Caution
Use cockpit only if you are an administrator of GNU/Linux systems with advanced skills. FreedomBox tries to coexist with changes to system by system administrators and system administration tools like Cockpit. However, improper changes to the system might causes failures in FreedomBox functions.

6.3.1 Using Cockpit

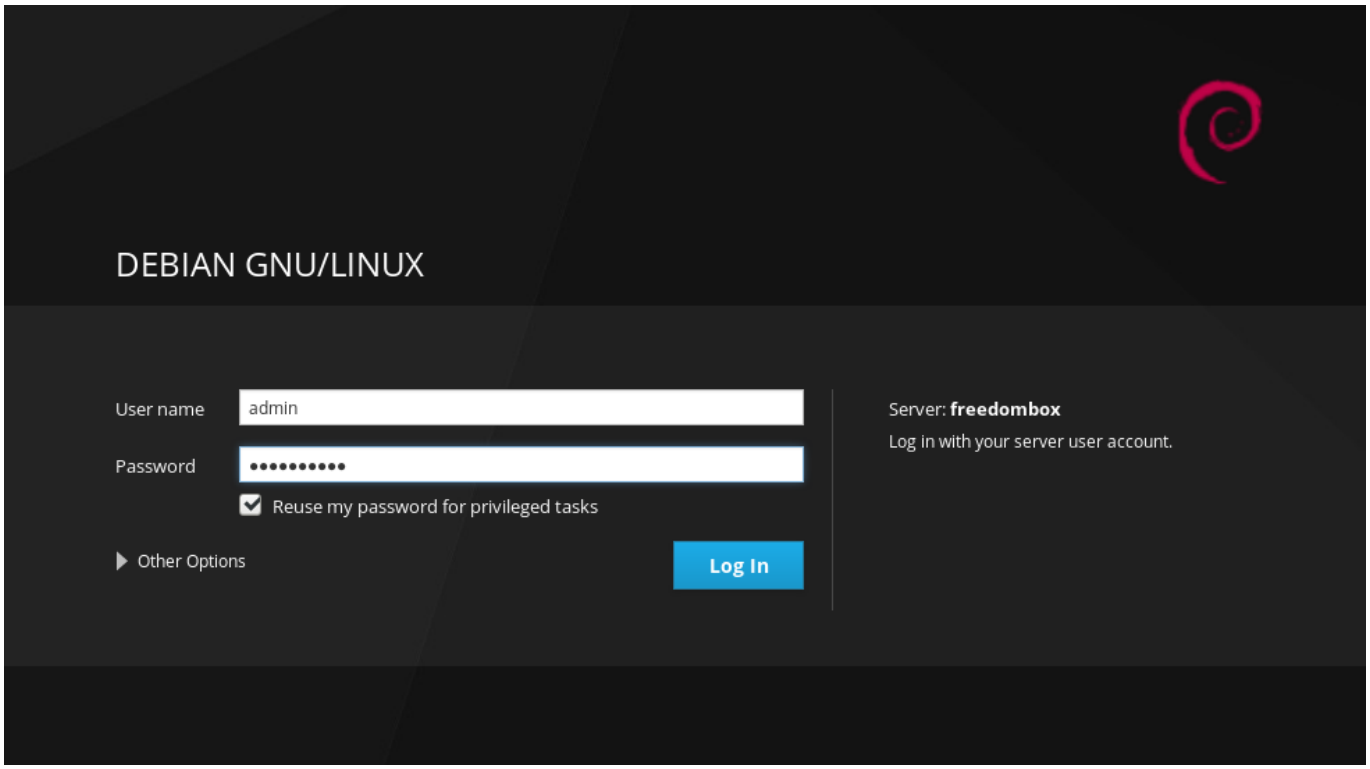
Install Cockpit like any other application on FreedomBox. Make sure that Cockpit is enabled after that.



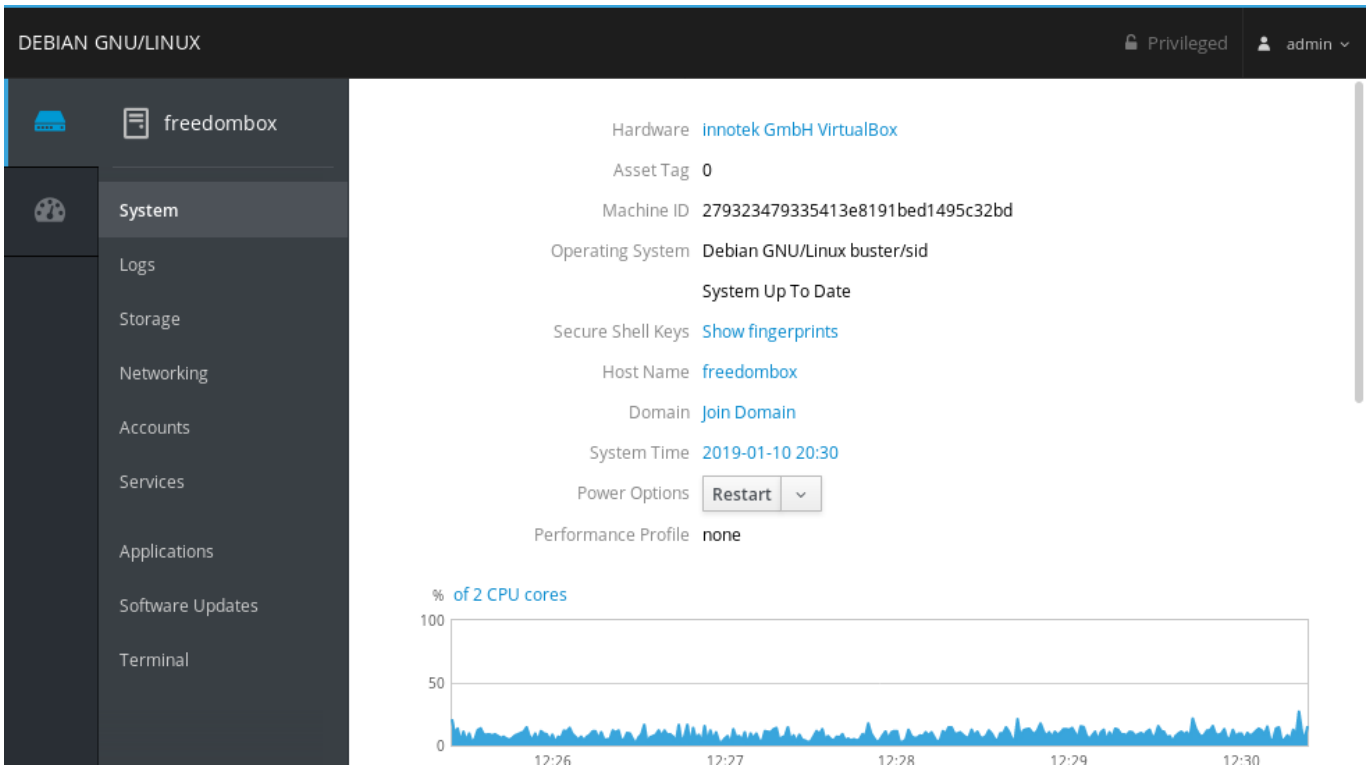
Ensure that the user account on FreedomBox that will used for Cockpit is part of the administrators group.



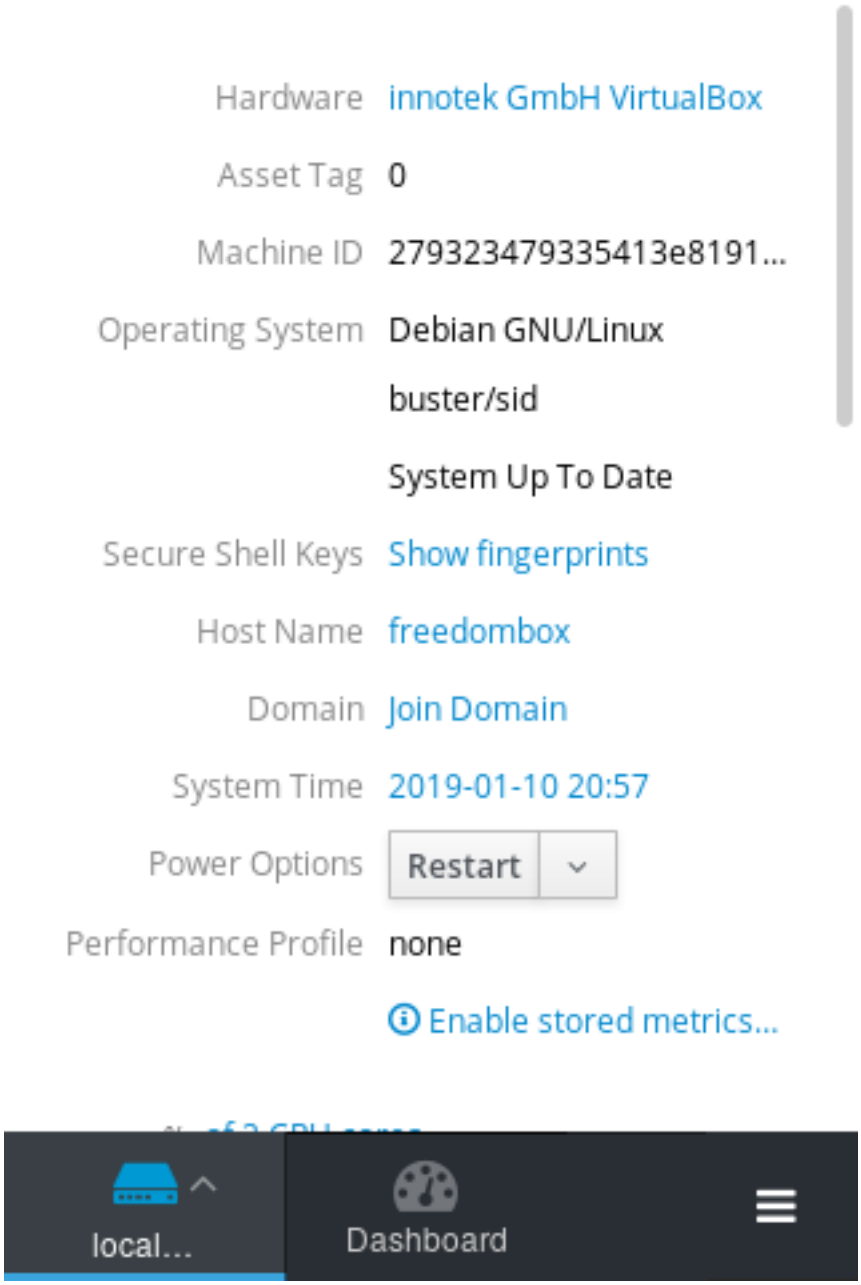
Launch the Cockpit web interface. Login using the configured user account.



Start using cockpit.



Cockpit is usable on mobile interfaces too.



6.3.2 Features

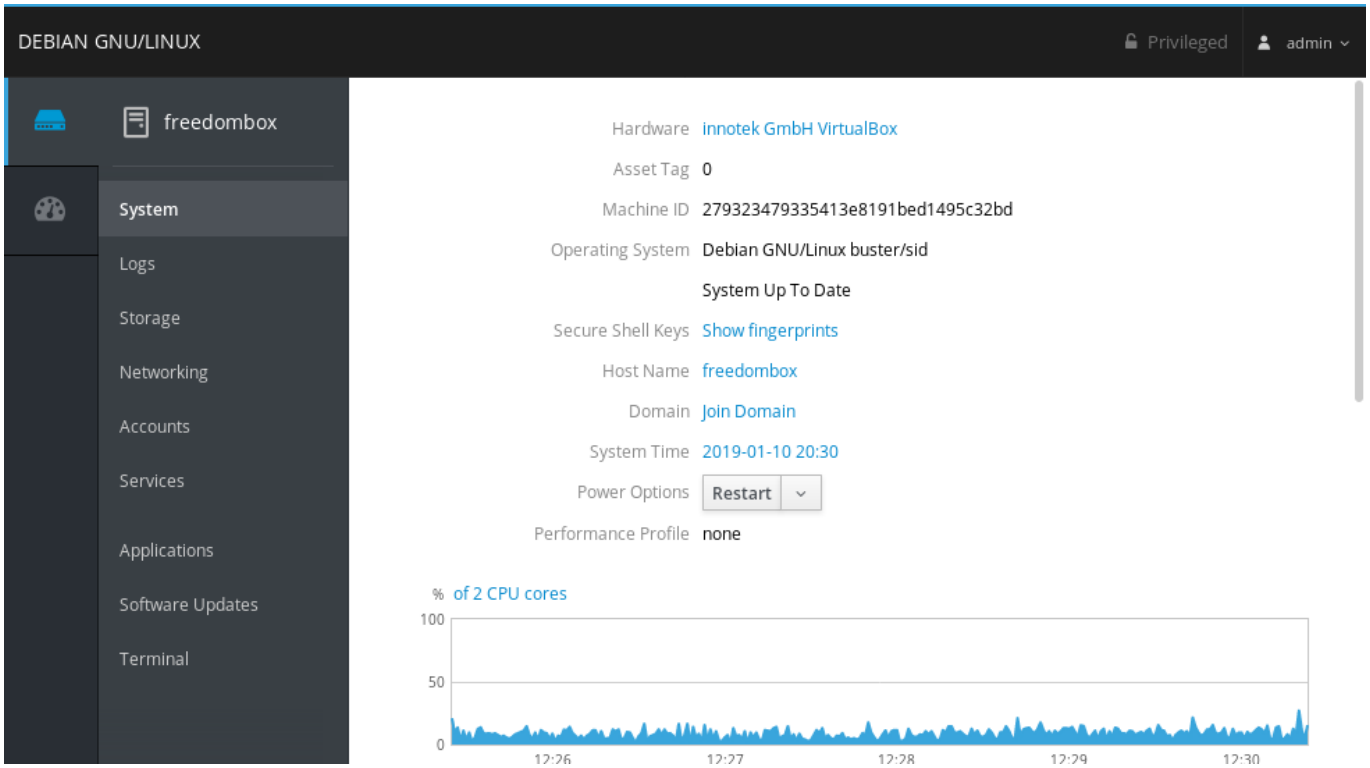
The following features of Cockpit may be useful for advanced FreedomBox users.

6.3.2.1 System Dashboard

Cockpit has a system dashboard that

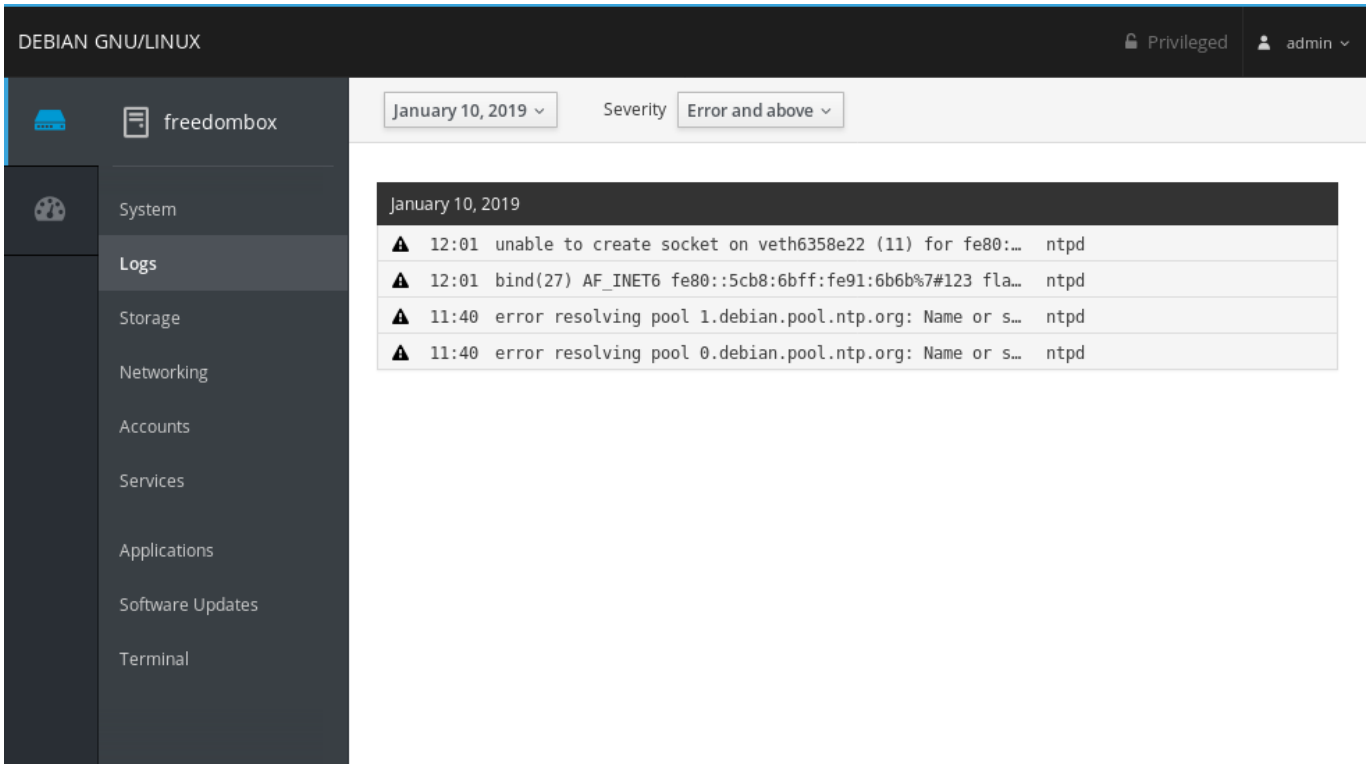
- Shows detailed hardware information
- Shows basic performance metrics of a system
- Allows changing system time and timezone
- Allows changing hostname. Please use FreedomBox UI to do this

- Shows SSH server fingerprints



6.3.2.2 Viewing System Logs

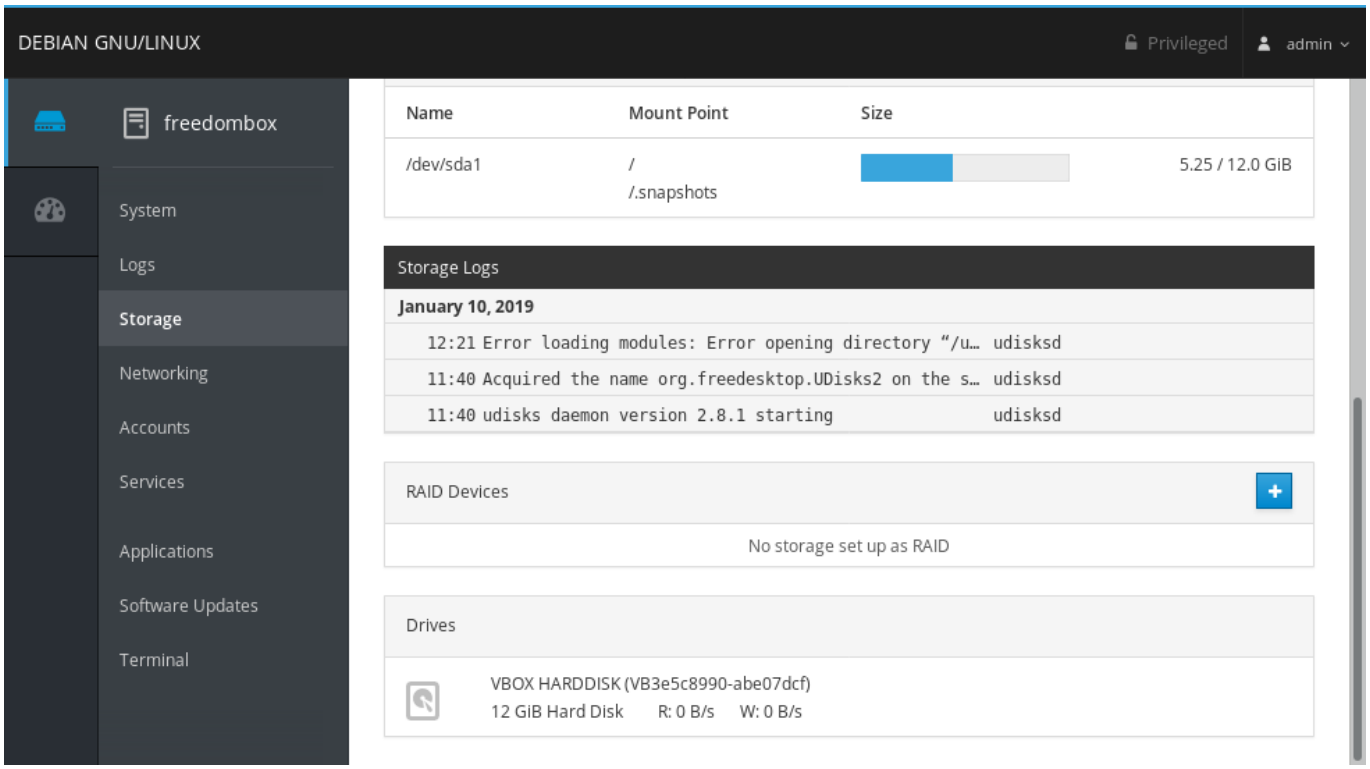
Cockpit allows querying system logs and examining them in full detail.

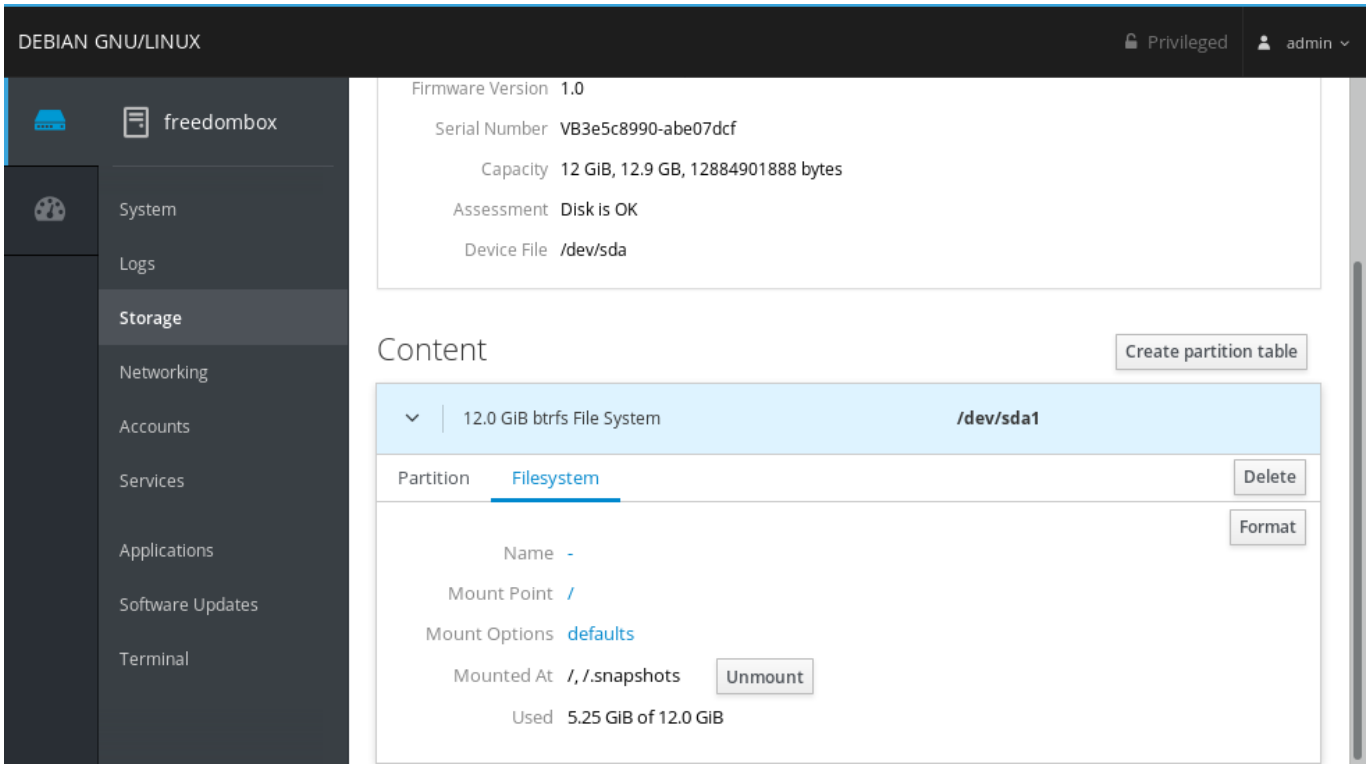


6.3.2.3 Managing Storage

Cockpit allows following advanced storage functions:

- View full disk information
- Editing disk partitions
- RAID management

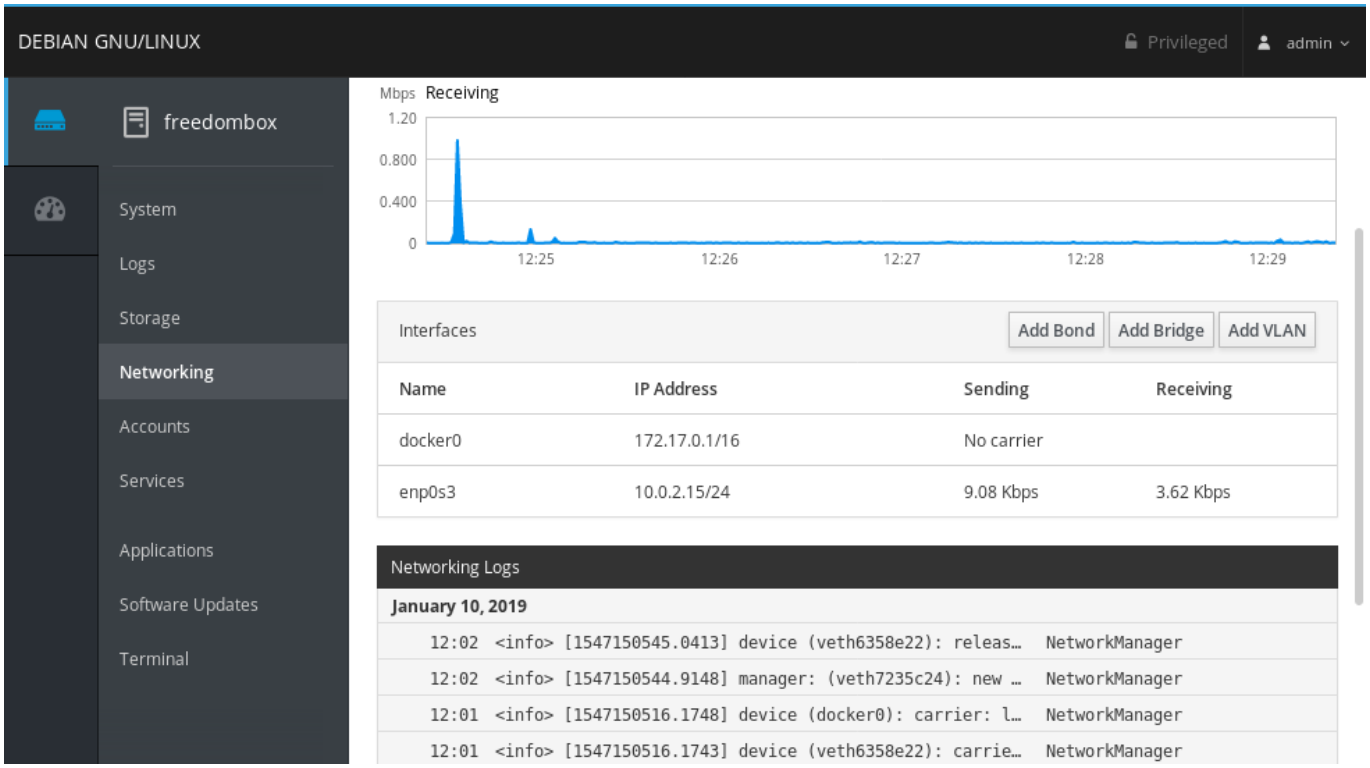


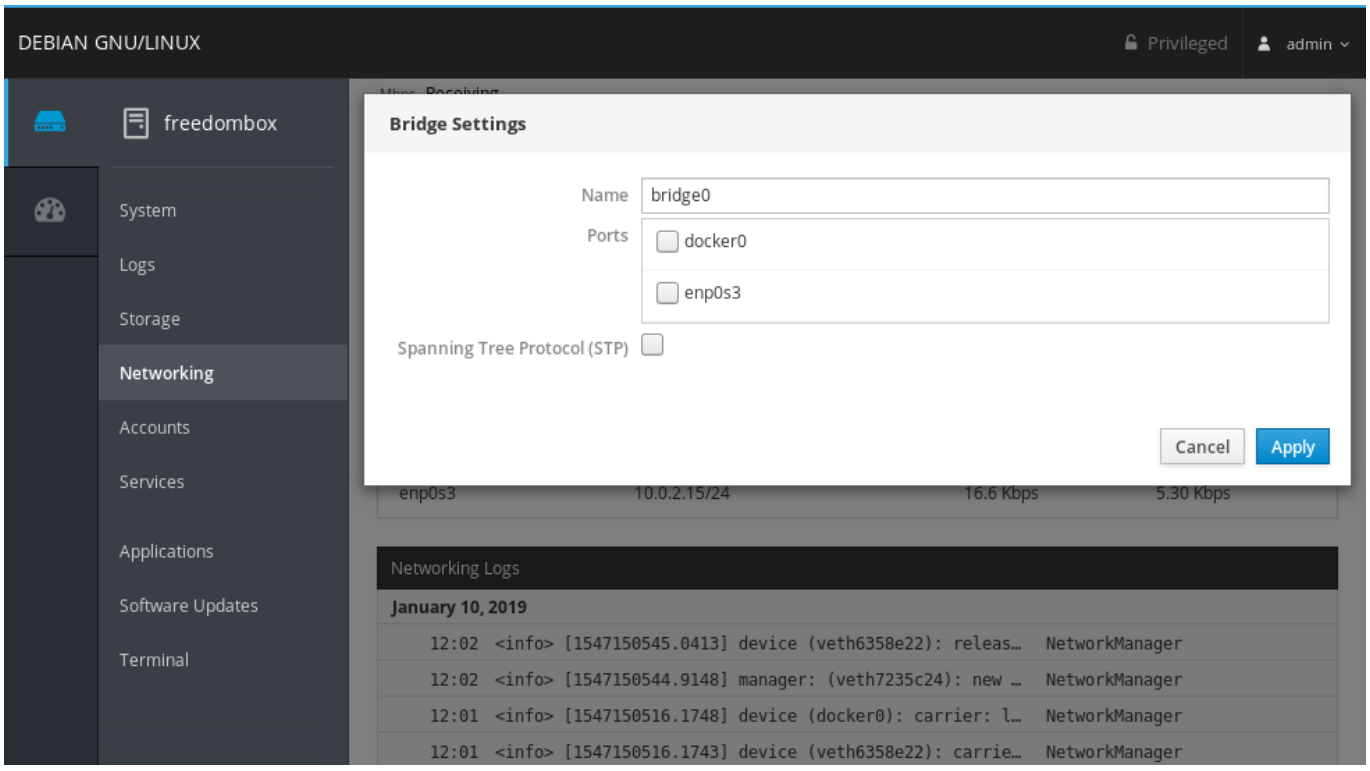


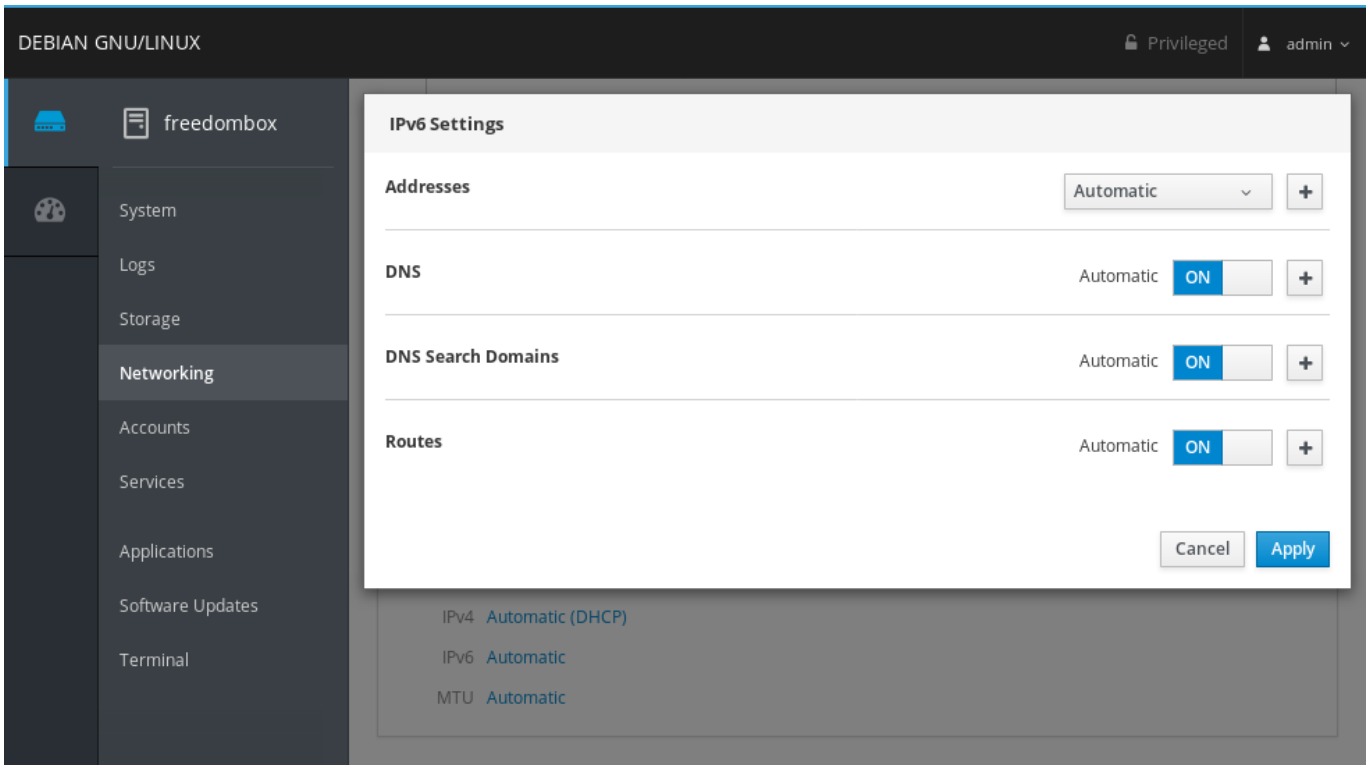
6.3.2.4 Networking

Cockpit and FreedomBox both rely on NetworkManager to configure the network. However, Cockpit offers some advanced configuration not available on FreedomBox:

- Route configuration
- Configure Bonds, Bridges, VLANs

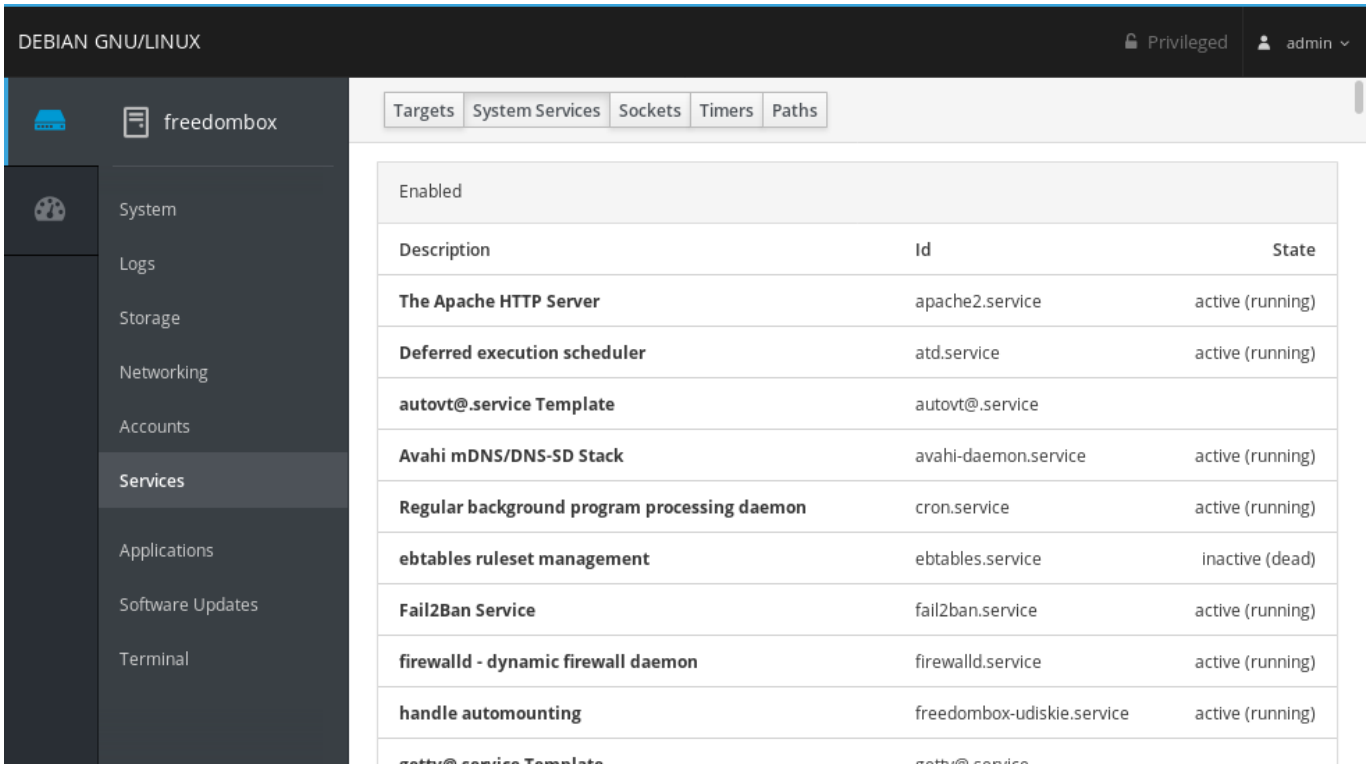


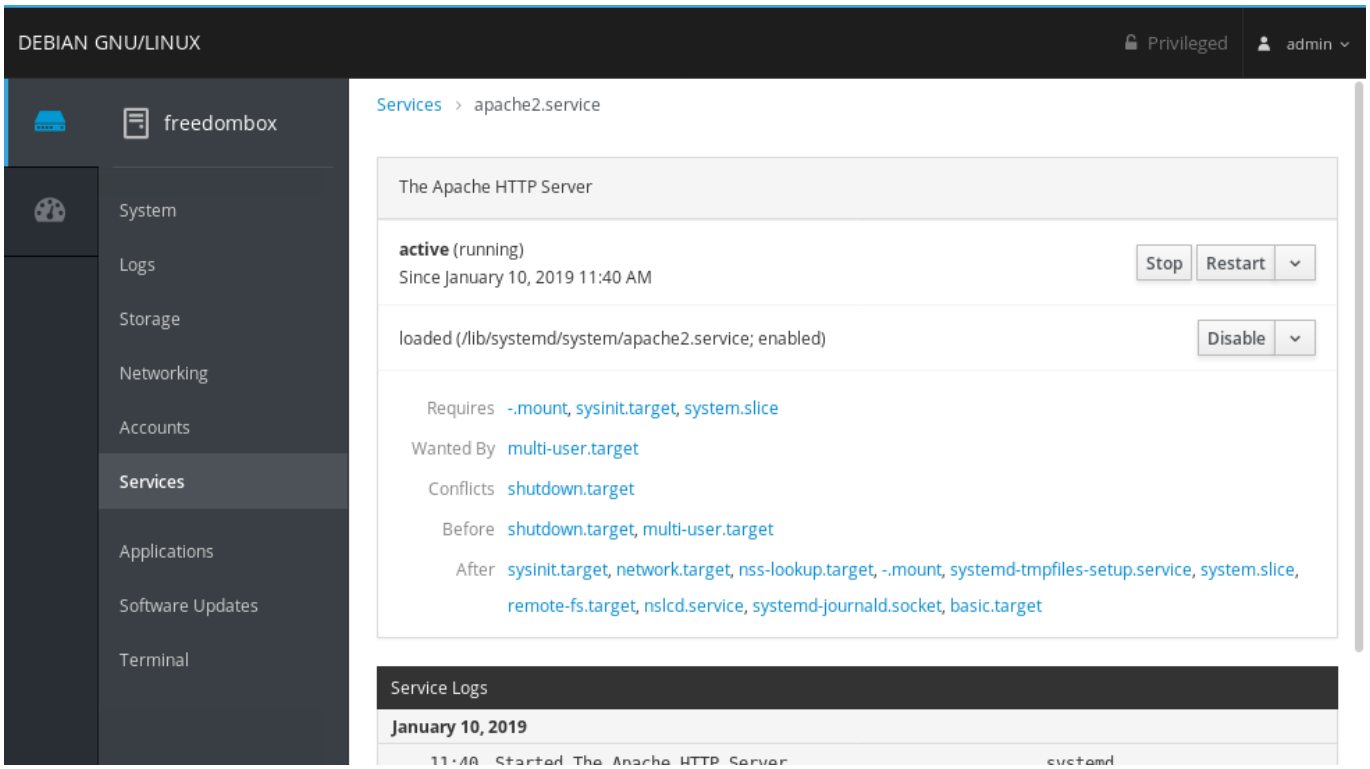




6.3.2.5 Services

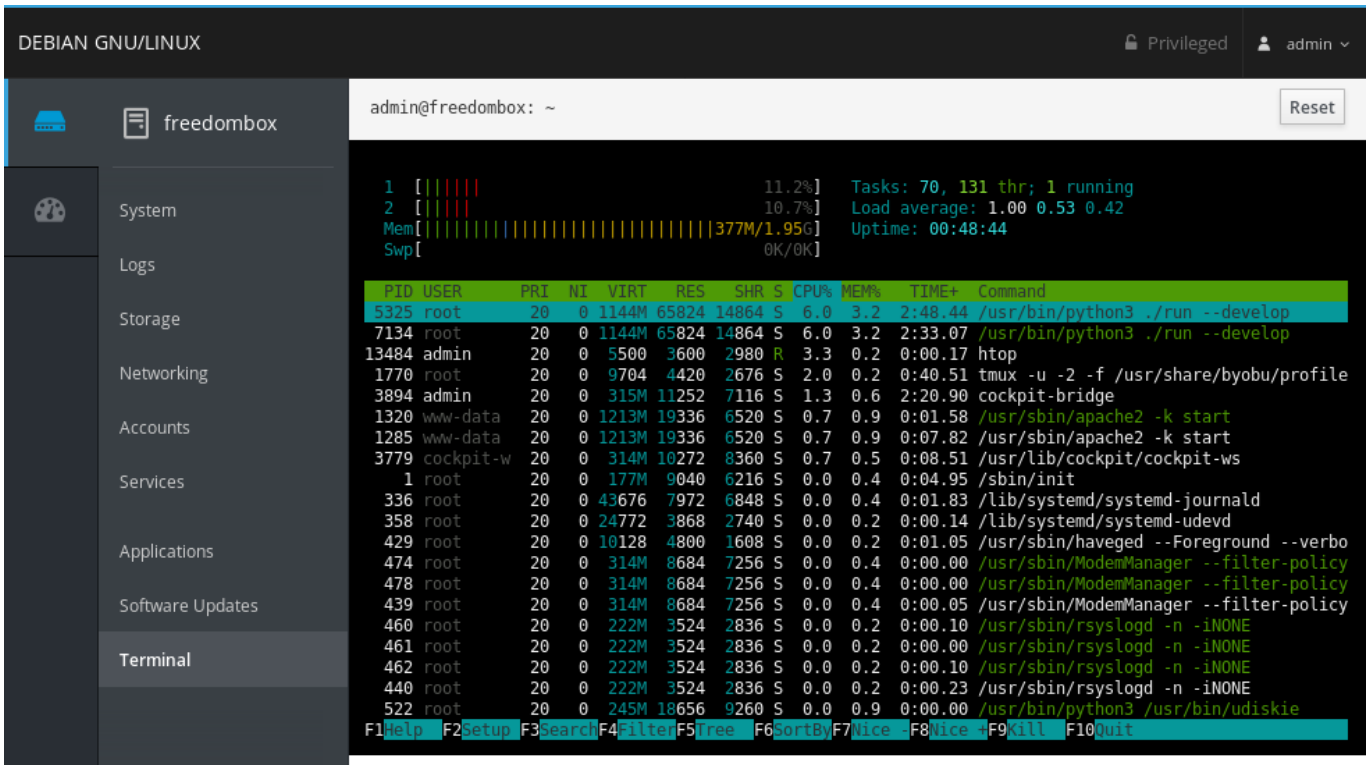
Cockpit allows management of services and periodic jobs (similar to cron).





6.3.2.6 Web Terminal

Cockpit offers a web based terminal that can be used perform manual system administration tasks.



6.3.3 Troubleshooting

Cockpit requires a domain name to be properly setup on your FreedomBox and will only work when you access it using a URL with that domain name. Cockpit will not work when using IP address in the URL. Using *freedombox.local* as the domain name also does not work. For example, the following URLs will not work:

```
https://192.168.0.10/_cockpit/
https://freedombox.local/_cockpit/
```

Starting with FreedomBox version 19.15, using *.local* domain works. You can access Cockpit using the URL `https://freedombox.local/_cockpit/`. The *.local* domain is based on your hostname. If your hostname is *mybox*, your *.local* domain name will be *mybox.local* and the Cockpit URL will be `https://mybox.local/_cockpit/`.

To properly access Cockpit, use the domain name configured for your FreedomBox.Cockpit will also work well when using a Tor Onion Service. The following URLs will work:

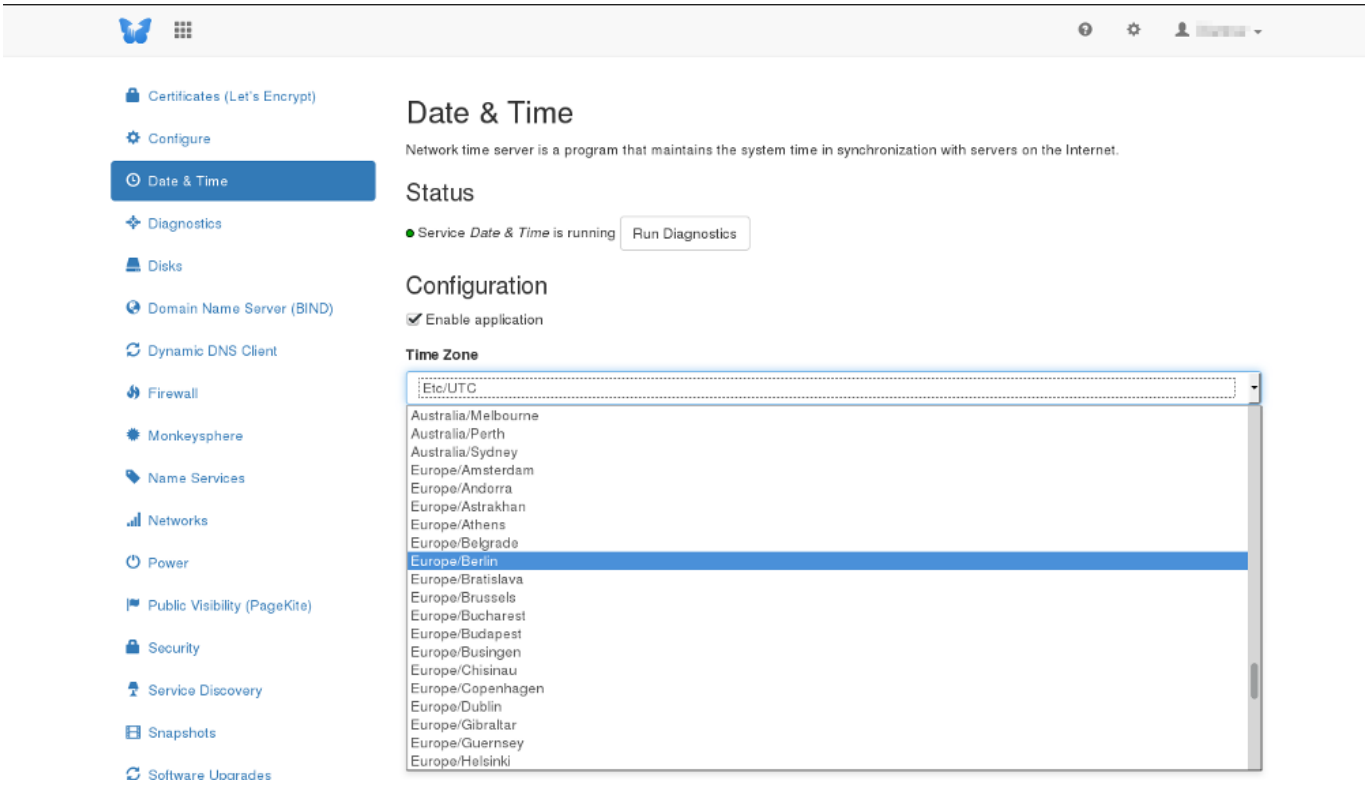
```
https://mybox.freedombox.rocks/_cockpit/
https://exampletorhs.onion/_cockpit/
```

The reason for this behaviour is that Cockpit uses WebSockets to connect to the backend server. Cross site requests for WebSockets must be prevented for security reasons. To implement this, Cockpit maintains a list of all domains from which requests are allowed. FreedomBox automatically configures this list whenever you add or remove a domain. However, since we can't rely on IP addresses, they are not added by FreedomBox to this domain list. You can see the current list of allowed domains, as managed by FreedomBox, in */etc/cockpit/cockpit.conf*. You may edit this, but do so only if you understand web security consequences of this.

6.4 Date & Time

This network time server is a program that maintains the system time in synchronization with servers on the Internet.

You can select your time zone by picking a big city nearby (they are sorted by *Continent/City*) or select directly the zone with respect to GMT (Greenwich Mean Time).



6.5 Diagnostics

The system diagnostic test will run a number of checks on your system to confirm that applications and services are working as expected.

Just click *Run Diagnostics*. This may take some minutes.

6.6 Dynamic DNS Client

6.6.1 What is Dynamic DNS?

In order to reach a server on the Internet, the server needs to have permanent address also known as the static IP address. Many Internet service providers don't provide home users with a static IP address or they charge more providing a static IP address. Instead they provide the home user with an IP address that changes every time the user connects to the Internet. Clients wishing to contact the server will have difficulty reaching the server.

Dynamic DNS service providers assist in working around a problem. First they provide you with a domain name, such as 'myhost.example.org'. Then they associate your IP address, whenever it changes, with this domain name. Then anyone intending to reach the server will be to contact the server using the domain name 'myhost.example.org' which always points to the latest IP address of the server.

For this to work, every time you connect to the Internet, you will have to tell your Dynamic DNS provider what your current IP address is. Hence you need special software on your server to perform this operation. The Dynamic DNS function in FreedomBox will allow users without a static public IP address to push the current public IP address to a Dynamic DNS Server. This allows you to expose services on FreedomBox, such as ownCloud, to the Internet.

6.6.2 GnuDIP vs. Update URL

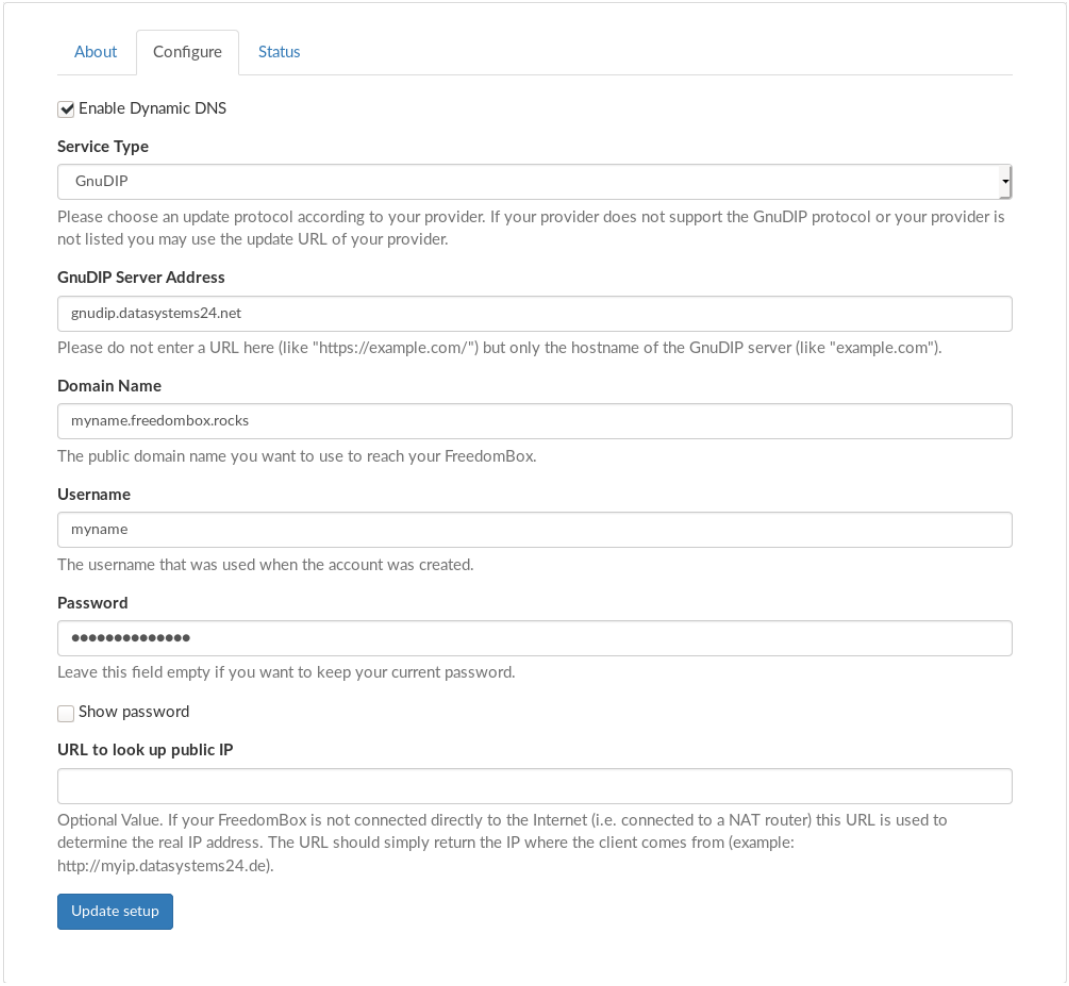
There are two main mechanism to notify the Dynamic DNS server of your new IP address; using the *GnuDIP* protocol and using the *Update URL* mechanism.

If a service provided using update URL is not properly secured using HTTPS, your credentials may be visible to an adversary. Once an adversary gains your credentials, they will be able to replay your request your server and hijack your domain.

On the other hand, the GnuDIP protocol will only transport a salted MD5 value of your password, in a way that is secure against replay attacks.

6.6.3 Using the GnuDIP protocol

1. Register an account with any Dynamic DNS service provider. A free service provided by the FreedomBox community is available at `https://gnudip.datasystems24.net` .
2. In FreedomBox UI, enable the Dynamic DNS Service.
3. Select *GnuDIP* as *Service type*, enter your Dynamic DNS service provider address (for example, gnudip.datasystems24.net) into *GnuDIP Server Address* field.



4. Fill *Domain Name*, *Username*, *Password* information given by your provider into the corresponding fields.

6.6.4 Using an Update URL

This feature is implemented because the most popular Dynamic DNS providers are using Update URLs mechanism.

1. Register an account with a Dynamic DNS service provider providing their service using Update URL mechanism. Some example providers are listed in the configuration page itself.
2. In FreedomBox UI, enable the Dynamic DNS service.
3. Select *other Update URL* as *Service type*, enter the update URL given by your provider into *Update URL* field.
4. If you browse the update URL with your Internet browser and a warning message about untrusted certificate appears, then enable *accept all SSL certificates*. WARNING: your credentials may be readable here because man-in-the-middle attacks are possible! Consider choosing a better service provider instead.
5. If you browse the update URL with your Internet browser and the username/password box appears, enable *use HTTP basic authentication* checkbox and provide the *Username* and *Password*.
6. If the update URL contains your current IP address, replace the IP address with the string *<Ip>*.

6.6.5 Checking If It Works

1. Make sure that external services you have enabled such as /jwchat, /roundcube and /ikiwiki are available on your domain address.

2. Go to the *Status* page, make sure that the NAT type is detected correctly. If your FreedomBox is behind a NAT device, this should be detected over there (Text: *Behind NAT*). If your FreedomBox has a public IP address assigned, the text should be "Direct connection to the Internet".
3. Check that the last update status is not *failed*.

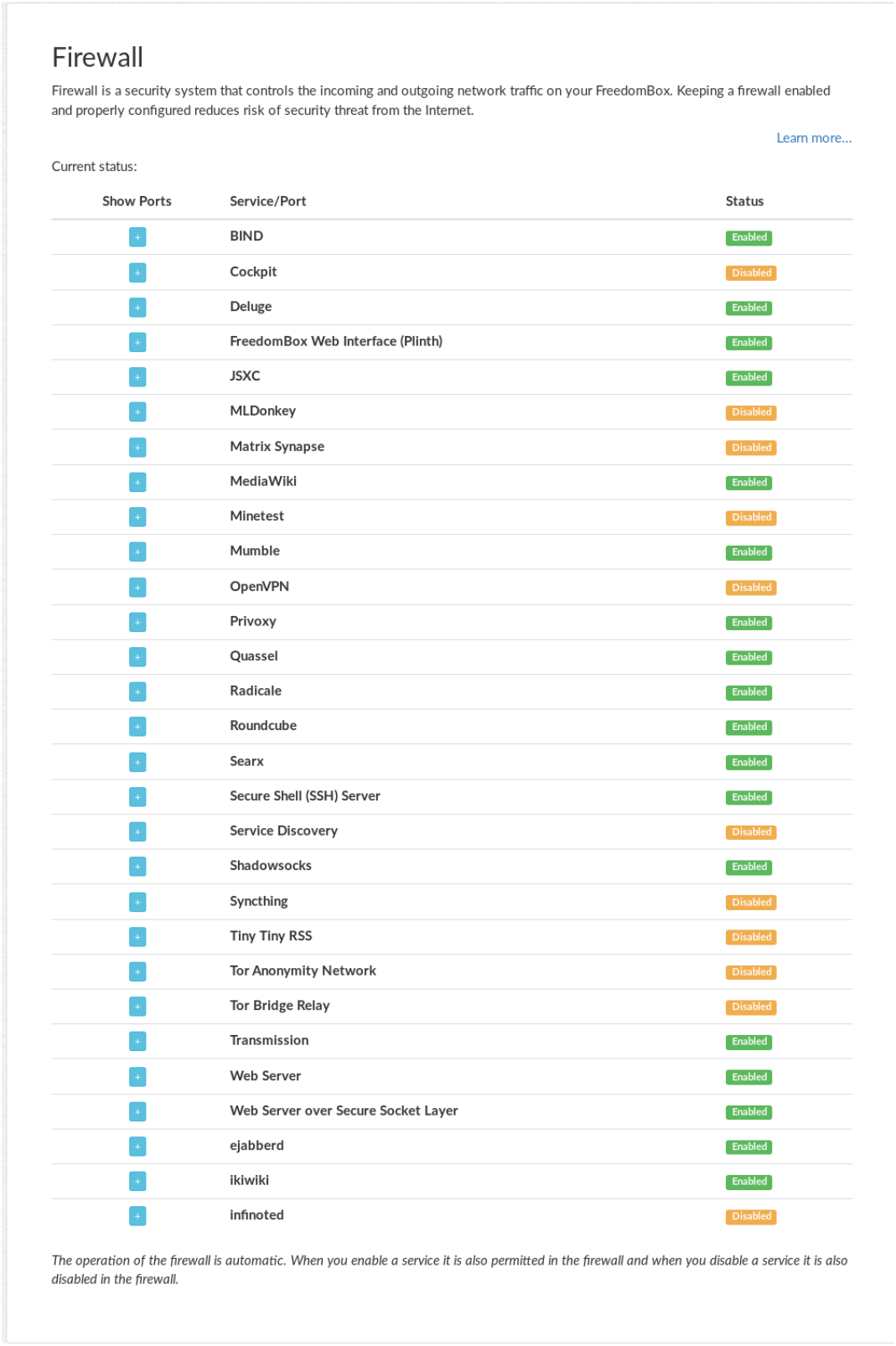
Recap: How to create a DNS name with GnuDIP

1. Access to GnuIP login page (answer Yes to all pop ups)
2. Click on "Self Register"
3. Fill the registration form (Username and domain will form the public IP address [username.domain])
4. Take note of the username/hostname and password that will be used on the FreedomBox app.
5. Save and return to the GnuDIP login page to verify your username, domain and password (enter the datas, click login).
6. Login output should display your new domain name along with your current public IP address (this is a unique address provided by your router for all your local devices).
7. Leave the GnuDIP interface and open the Dynamic DNS Client app page in your FreedomBox.
8. Click on "Set Up" in the top menu.
9. Activate Dynamic DNS
10. Choose GnuDIP service.
11. Add server address (gnudip.datasystems24.net)
12. Add your fresh domain name (username.domain, ie [username].freedombox.rocks)
13. Add your fresh username (the one used in your new IP address) and password
14. Add your GnuDIP password
15. Fill the option with `http://myip.datasystems24.de` (try this url in your browser, you will figure out immediately)

6.7 Firewall

Firewall is a network security system that controls the incoming and outgoing network traffic. Keeping a firewall enabled and properly configured reduces risk of security threat from the Internet.

The operation of the firewall in Plinth web interface of FreedomBox is automatic. When you enable a service it is automatically permitted in the firewall and when you disable a service it is automatically disabled in the firewall. For services which are enabled by default on FreedomBox, firewall ports are also enabled by default during the first run process.



Firewall

Firewall is a security system that controls the incoming and outgoing network traffic on your FreedomBox. Keeping a firewall enabled and properly configured reduces risk of security threat from the Internet.

Learn more...

Current status:

Show Ports	Service/Port	Status
+	BIND	Enabled
+	Cockpit	Disabled
+	Deluge	Enabled
+	FreedomBox Web Interface (Plinth)	Enabled
+	JSXC	Enabled
+	MLDonkey	Disabled
+	Matrix Synapse	Disabled
+	MediaWiki	Enabled
+	Minetest	Disabled
+	Mumble	Enabled
+	OpenVPN	Disabled
+	Privoxy	Enabled
+	Quassel	Enabled
+	Radicale	Enabled
+	Roundcube	Enabled
+	Searx	Enabled
+	Secure Shell (SSH) Server	Enabled
+	Service Discovery	Disabled
+	Shadowsocks	Enabled
+	Syncthing	Disabled
+	Tiny Tiny RSS	Disabled
+	Tor Anonymity Network	Disabled
+	Tor Bridge Relay	Disabled
+	Transmission	Enabled
+	Web Server	Enabled
+	Web Server over Secure Socket Layer	Enabled
+	ejabberd	Enabled
+	ikiwiki	Enabled
+	infinoted	Disabled

The operation of the firewall is automatic. When you enable a service it is also permitted in the firewall and when you disable a service it is also disabled in the firewall.

Firewall management in FreedomBox is done using FirewallD.

6.7.1 Interfaces

Each interface is needs to be assigned to one (and only one) zone. Whatever rules are in effect for a zone, those rules start to apply for that interface. For example, if HTTP traffic is allowed in a particular zone, then web requests will be accepted on all

the addresses configured for all the interfaces assigned to that zone.

There are primarily two firewall zones used. The `internal` zone is meant for services that are provided to all machines on the local network. This may include services such as streaming media and simple file sharing. The `external` zone is meant for services that are provided publicly on the Internet. This may include services such as blog, website, email web client etc.

For details on how network interfaces are configured by default, see the Networks section.

6.7.2 Ports/Services

The following table attempts to document the ports, services and their default statuses in FreedomBox. If you find this page outdated, see the Plinth source for lib/freedombox/first-run.d/90_firewall and Firewall status page in Plinth UI.

Service	Port	External	Enabled by default	Status shown in Plinth	Managed by Plinth
Minetest	30000/udp	☆	✗	✓	✓
XMPP Client	5222/tcp	☆	✗	✓	✓
XMPP Server	5269/tcp	☆	✗	✓	✓
XMPP Bosh	5280/tcp	☆	✗	✓	✓
NTP	123/udp	☆	✓	✓	✓
Plinth	443/tcp	☆	✓	✓	✗
Quassel	4242/tcp	☆	✗	✓	✓
SIP	5060/tcp	☆	✗	✓	✓
SIP	5060/udp	☆	✗	✓	✓
SIP-TLS	5061/tcp	☆	✗	✓	✓
SIP-TLS	5061/udp	☆	✗	✓	✓
RTP	1024-65535/udp	☆	✗	✓	✓
SSH	22/tcp	☆	✓	✓	✗
mDNS	5353/udp	☆	✓	✓	✓
Tor (Socks)	9050/tcp	☆	✗	✓	✓
Obfsproxy	<random>/tcp	☆	✗	✓	✓
OpenVPN	1194/udp	☆	✗	✓	✓
Mumble	64378/tcp	☆	✗	✓	✓
Mumble	64378/udp	☆	✗	✓	✓
Privoxy	8118/tcp	☆	✗	✓	✓
JSXC	80/tcp	☆	✗	✗	✗
JSXC	443/tcp	☆	✗	✗	✗
DNS	53/tcp	☆	✗	✗	✗
DNS	53/udp	☆	✗	✗	✗
DHCP	67/udp	☆	✓	✗	✗
Bootp	67/tcp	☆	✗	✗	✗
Bootp	67/udp	☆	✗	✗	✗
Bootp	68/tcp	☆	✗	✗	✗
Bootp	68/udp	☆	✗	✗	✗
LDAP	389/tcp	☆	✗	✗	✗
LDAPS	636/tcp	☆	✗	✗	✗

6.7.3 Manual operation

See FirewallD documentation for more information on the basic concepts and comprehensive documentation.

6.7.3.1 Enable/disable firewall

To disable firewall

```
service firewalld stop
```

or with systemd

```
systemctl stop firewalld
```

To re-enable firewall

```
service firewalld start
```

or with systemd

```
systemctl start firewalld
```

6.7.3.2 Modifying services/ports

You can manually add or remove a service from a zone.

To see list of services enabled:

```
firewall-cmd --zone=<zone> --list-services
```

Example:

```
firewall-cmd --zone=internal --list-services
```

To see list of ports enabled:

```
firewall-cmd --zone=<zone> --list-ports
```

Example:

```
firewall-cmd --zone=internal --list-ports
```

To remove a service from a zone:

```
firewall-cmd --zone=<zone> --remove-service=<service>
firewall-cmd --permanent --zone=<zone> --remove-service=<interface>
```

Example:

```
firewall-cmd --zone=internal --remove-service=xmpp-bosh
firewall-cmd --permanent --zone=internal --remove-service=xmpp-bosh
```

To remove a port from a zone:

```
firewall-cmd --zone=internal --remove-port=<port>/<protocol>
firewall-cmd --permanent --zone=internal --remove-port=<port>/<protocol>
```

Example:

```
firewall-cmd --zone=internal --remove-port=5353/udp
firewall-cmd --permanent --zone=internal --remove-port=5353/udp
```

To add a service to a zone:

```
firewall-cmd --zone=<zone> --add-service=<service>
firewall-cmd --permanent --zone=<zone> --add-service=<interface>
```

Example:

```
firewall-cmd --zone=internal --add-service=xmpp-bosh
firewall-cmd --permanent --zone=internal --add-service=xmpp-bosh
```

To add a port to a zone:

```
firewall-cmd --zone=internal --add-port=<port>/<protocol>
firewall-cmd --permanent --zone=internal --add-port=<port>/<protocol>
```

Example:

```
firewall-cmd --zone=internal --add-port=5353/udp
firewall-cmd --permanent --zone=internal --add-port=5353/udp
```

6.7.3.3 Modifying the zone of interfaces

You can manually change the assignment of zones of each interfaces after they have been autuomatically assigned by the first boot process.

To see current assignment of interfaces to zones:

```
firewall-cmd --list-all-zones
```

To remove an interface from a zone:

```
firewall-cmd --zone=<zone> --remove-interface=<interface>
firewall-cmd --permanent --zone=<zone> --remove-interface=<interface>
```

Example:

```
firewall-cmd --zone=external --remove-interface=eth0
firewall-cmd --permanent --zone=external --remove-interface=eth0
```

To add an interface to a zone:

```
firewall-cmd --zone=<zone> --add-interface=<interface>
firewall-cmd --permanent --zone=<zone> --add-interface=<interface>
```

Example:

```
firewall-cmd --zone=internal --add-interface=eth0
firewall-cmd --permanent --zone=internal --add-interface=eth0
```

6.8 Certificates (Let's Encrypt)

A digital certificate allows users of a web service to verify the identity of the service and to securely communicate with it. FreedomBox can automatically obtain and setup digital certificates for each available domain. It does so by proving itself to be the owner of a domain to Let's Encrypt, a certificate authority (CA).

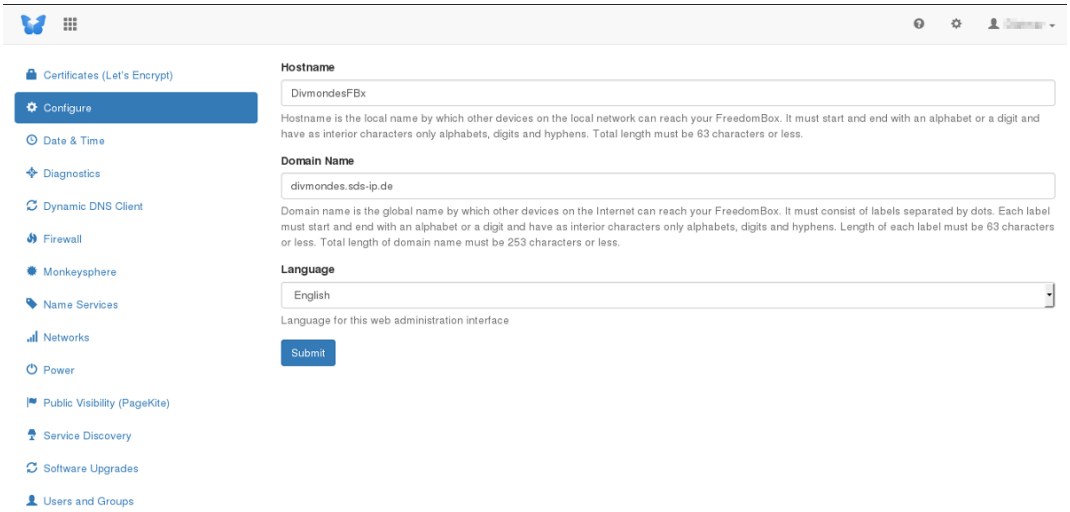
Let's Encrypt is a free, automated, and open certificate authority, run for the public's benefit by the Internet Security Research Group (ISRG). Please read and agree with the Let's Encrypt Subscriber Agreement before using this service.

6.8.1 Why using Certificates

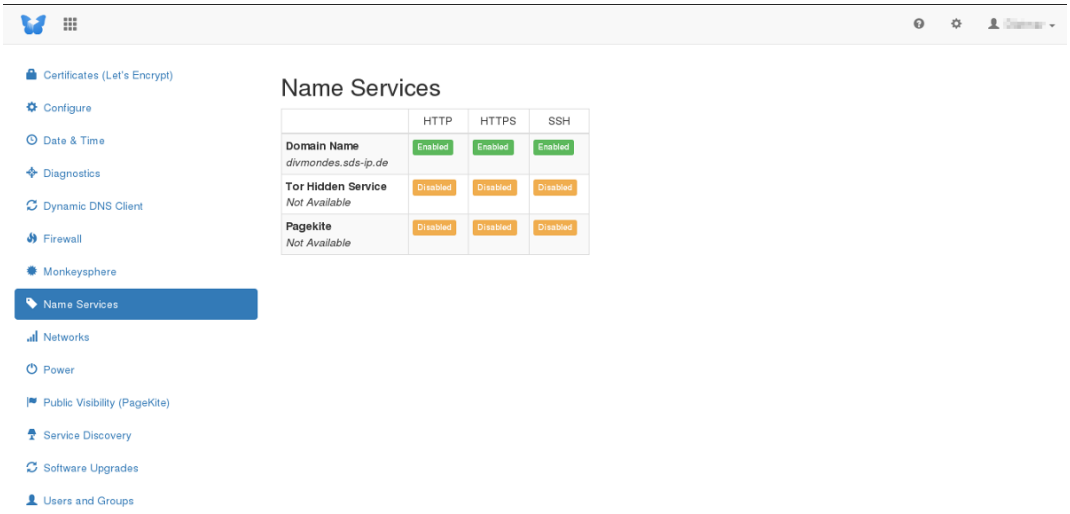
The communication with your FreedomBox can be secured so that it is not possible to intercept the content of the web pages viewed and about the content exchanged.

6.8.2 How to setup

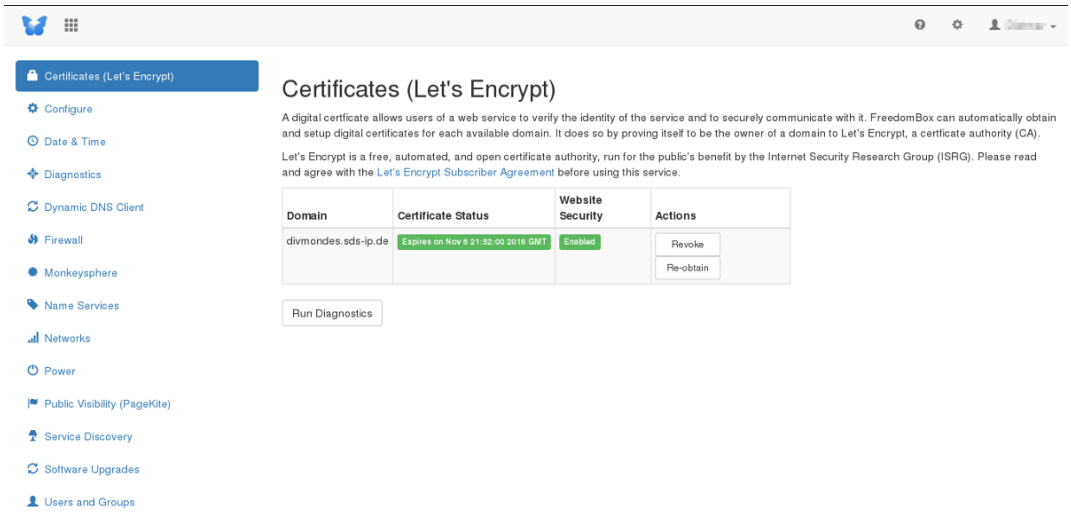
1. If your FreedomBox is behind a router, you will need to set up port forwarding on your router. You should forward the following ports:
   - TCP 80 (http)
   - TCP 443 (https)
2. Make the domain name known:
   - In Configure insert your *domain name*, e.g. `MyWebName.com`



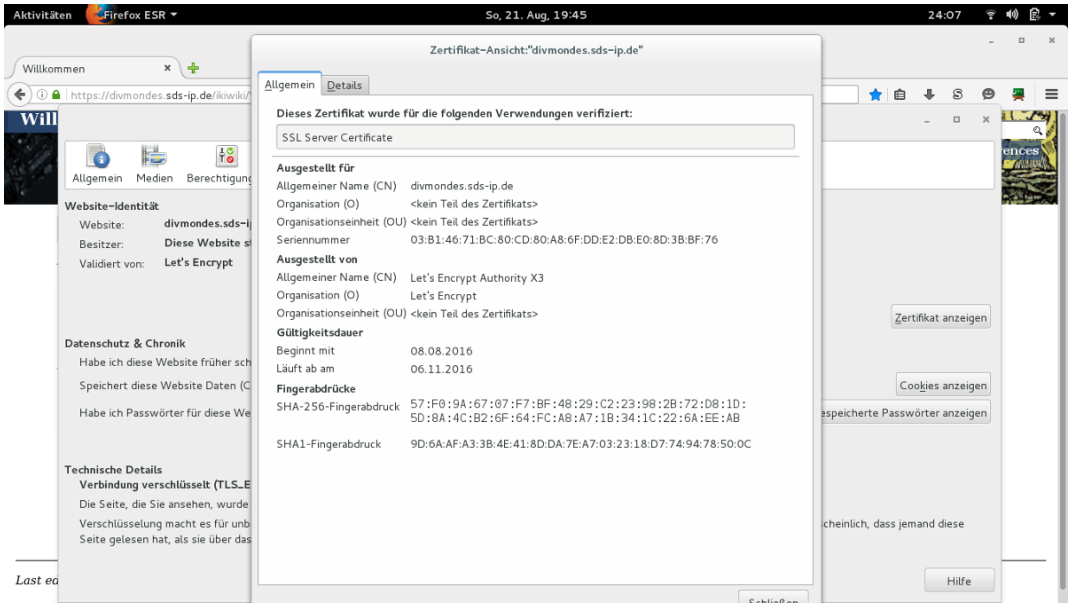
3. Verify the domain name was accepted
   - Check that it is enabled in Name Services



4. Go to the Certificates (Let's Encrypt) page, and complete the module install if needed. Then click the "Obtain" button for your domain name.
   - After some minutes a valid certificate is available



5. Verify in your browser by checking `https://MyWebName.com`



Screencast: Let's Encrypt

6.8.3 Using

The certificate is valid for 3 months. It is renewed automatically and can also be re-obtained or revoked manually.

With running *diagnostics* the certificate can also be verified.

6.9 Monkeysphere

With Monkeysphere, an OpenPGP key can be generated for each configured domain serving SSH. The OpenPGP public key can then be uploaded to the OpenPGP keyservers. Users connecting to this machine through SSH can verify that they are connecting to the correct host. For users to trust the key, at least one person (usually the machine owner) must sign the key using the regular OpenPGP key signing process. See the Monkeysphere SSH documentation for more details.

Monkeysphere can also generate an OpenPGP key for each Secure Web Server (HTTPS) certificate installed on this machine. The OpenPGP public key can then be uploaded to the OpenPGP keyservers. Users accessing the web server through HTTPS can verify that they are connecting to the correct host. To validate the certificate, the user will need to install some software that is available on the Monkeysphere website.

6.10 Name Services

Name Services provides an overview of ways the box can be reached from the public Internet: domain name, Tor Onion Service, and Pagekite. For each type of name, it is shown whether the HTTP, HTTPS, and SSH services are enabled or disabled for incoming connections through the given name.

6.11 Networks

This section describes how networking is setup by default in FreedomBox and how you can customize it. See also the Firewall section for more information on how firewall works.

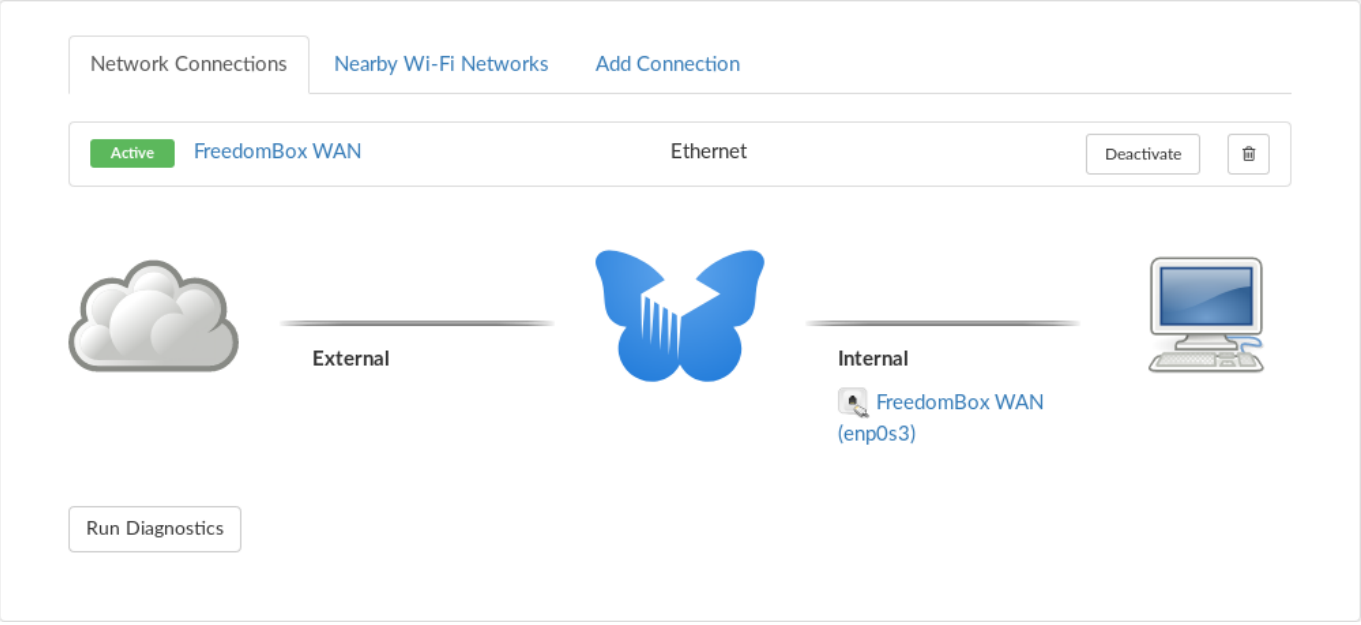
6.11.1 Default setup

In a fresh image of FreedomBox, network is not configured at all. When the image is written to an SD card and the device boots, configuration is done. During first boot, FreedomBox setup package detects the networks interfaces and tries to automatically configure them so that FreedomBox is available for further configuration via the web interface from another machine without the need to connect a monitor. Automatic configuration also tries to make FreedomBox useful, out of the box, for the most important scenarios FreedomBox is used for.

There are two scenarios it handles: when is a single ethernet interface and when there are multiple ethernet interfaces.

6.11.1.1 Single ethernet interface

When there is only single ethernet interface available on the hardware device, there is not much scope for it to play the role of a router. In this case, the device is assumed to be just another machine in the network. Accordingly, the only available interface is configured to be an `internal` interface in `automatic` configuration mode. This means that it connects to the Internet using the configuration provided by a router in the network and also makes all (internal and external) of its services available to all the clients on this network.



6.11.1.2 Multiple ethernet interface

When there are multiple ethernet interfaces available on the hardware device, the device can act as a router. The interfaces are then configured to perform this function.

The first network interface is configured to be an WAN or `external` interface in automatic configuration mode. This means that it connects to the Internet using network configuration provided by the Internet Service Provider (ISP). Only services that are

meant to be provided across the entire Internet (external services) will be exposed on this interface. You must plug your Internet connection into the port of this ethernet interface. If you wish to continue to have your existing router manage the Internet connection for you, then plug a connection from your router to the port on this interface.

The remaining network interfaces are configured for the clients of a router. They are configured as LAN or `internal` interfaces in `shared` configuration mode. This means that all the services (both external and internal) services are provided to who ever connects on this interface. Further, the `shared` mode means that clients will be able to receive details of automatic network connection on this interface. Specifically, DHCP configuration and DNS servers are provided on this interface. The Internet connection available to the device using the first network interface will be `shared` with clients using this interface. This all means that you can connect your computers to this network interface and they will get automatically configured and will be able to access the Internet via the FreedomBox.

Currently, it is not very clear which interface will be come the WAN interface (and the remaining being LAN interfaces) although the assignment process is deterministic. So, it take a bit of trail and error to figure out which one is which. In future, for each device, this will be well documented.

6.11.1.3 Wi-Fi configuration

All Wi-Fi interfaces are configured to be LAN or `internal` interfaces in `shared` configuration mode. They are also configured to become Wi-Fi access points with following details.

- Name of the access point will be `FreedomBox` plus the name of the interface (to handle the case where there are multiple of them).
- Password for connecting to the interface will be `freedombox123`.

6.11.2 Internet Connection Sharing

Although the primary duty of FreedomBox is to provide decentralized services, it can also act like a home router. Hence, in most cases, FreedomBox connects to the Internet and provides other machines in the network the ability to use that Internet connection. FreedomBox can do this in two ways: using a `shared` mode connection or using an `internal` connection.

When an interface is set in `shared` mode, you may connect your machine directly to it. This is either by plugging in an ethernet cable from this interface to your machine or by connecting to a Wi-Fi access point. This case is the simplest to use, as FreedomBox automatically provides your machine with the necessary network configuration. Your machine will automatically connect to FreedomBox provided network and will be able to connect to the Internet given that FreedomBox can itself connect to the Internet.

Sometimes the above setup may not be possible because the hardware device may have only one network interface or for other reasons. Even in this case, your machine can still connect to the Internet via FreedomBox. For this to work, make sure that the network interface that your machine is connecting to is in `internal` mode. Then, connect your machine to network in which FreedomBox is present. After this, in your machine's network configuration, set FreedomBox's IP address as the gateway. FreedomBox will then accept your network traffic from your machine and send it over to the Internet. This works because network interfaces in `internal` mode are configured to `masquerade` packets from local machines to the Internet and receive packets from Internet and forward them back to local machines.

6.11.3 Customization

The above default configuration may not be fit for your setup. You can customize the configuration to suit your needs from the `Networks` area in the 'setup' section of the FreedomBox web interface.

6.11.3.1 PPPoE connections

If your ISP does not provide automatic network configuration via DHCP and requires you to connection via PPPoE. To configure PPPoE, remove any network connection existing on an interface and add a PPPoE connection. Here, optionally, provide the account username and password given by your ISP and activate the connection.

6.11.3.2 Connect to Internet via Wi-Fi

By default Wi-Fi devices attached during first boot will be configured as access points. They can be configured as regular Wi-Fi devices instead to connection to a local network or an existing Wi-Fi router. To do this, click on the Wi-Fi connection to edit it. Change the mode to `Infrastructure` instead of `Access Point` mode and `IPv4 Addressing Method` to `Automatic (DHCP)` instead of `Shared` mode. Then the SSID provided will mean the Wi-Fi network name you wish to connect to and passphrase will be the used to while making the connection.

6.11.3.2.1 Problems with Privacy Feature

NetworkManager used by FreedomBox to connect to the Wi-Fi networks has a privacy feature that uses a different identity when scanning for networks and when actually connecting to the Wi-Fi access point. Unfortunately, this causes problems with some routers that reject connections from such devices. Your connection won't successfully activate and disconnect after trying to activate. If you have control over the router's behaviour, you could also turn off the feature causing problem. Otherwise, the solution is to connect with a remote shell using SSH or Cockpit, editing a file `/etc/NetworkManager/NetworkManager.conf` and adding the line `wifi.scan-rand-mac-address=no` in the `[device]` section. This turns off the privacy feature.

Edit a file:

```
$ sudo nano /etc/NetworkManager/NetworkManager.conf
```

Add the following:

```
[device]
wifi.scan-rand-mac-address=no
```

Then reboot the machine.

6.11.3.3 Adding a new network device

When a new network device is added, network manager will automatically configure it. In most cases this will not work to your liking. Delete the automatic configuration created on the interface and create a new network connection. Select your newly added network interface in the add connection page.

- Then set firewall zone to `internal` and `external` appropriately.
- You can configure the interface to connect to a network or provide network configuration to whatever machine connects to it.
- Similarly, if it is a Wi-Fi interface, you can configure it to become a Wi-FI access point or to connect to an existing access points in the network.

6.11.3.4 Configuring a mesh network

FreedomBox has rudimentary support for participating in BATMAN-Adv based mesh networks. It is possible to either join an existing network in your area or create a new mesh network and share your Internet connection with the rest of the nodes that join the network. Currently, two connections have to be created and activated manually to join or create a mesh network.

6.11.3.4.1 Joining a mesh network

To join an existing mesh network in your area, first consult the organizers and get information about the mesh network.

1. Create a new connection, then select the connection type as *Wi-Fi*. In the following dialog, provide the following values:

Field Name	Example Value	Explanation
Connection Name	Mesh Join - BATMAN	The name must end with 'BATMAN' (uppercase)
Physical Interface	wlan0	The Wi-Fi device you wish to use for joining the mesh network
Firewall Zone	External	Since you don't wish that participants in mesh network to use internal services of FreedomBox
SSID	ch1.freifunk.net	As provided to you by the operators of the mesh network. You should see this as a network in *Nearby Wi-Fi Networks*
Mode	Ad-hoc	Because this is a peer-to-peer network
Frequency Band	2.4Ghz	As provided to you by the operators of the mesh network
Channel	1	As provided to you by the operators of the mesh network
BSSID	12:CA:FF:EE:BA:BE	As provided to you by the operators of the mesh network
Authentication	Open	Leave this as open, unless you know your mesh network needs it be otherwise
Passphrase		Leave empty unless you know your mesh network requires one
IPv4 Addressing Method	Disabled	We don't want to request IP configuration information yet

Save the connection. Join the mesh network by activating this newly created connection.

2. Create a second new connection, then select the connection type as *Generic*. In the following dialog, provide this following values:

Field Name	Example Value	Explanation
Connection Name	Mesh Connect	Any name to identify this connection
Physical Interface	bat0	This interface will only show up after you successfully activate the connection in first step
Firewall Zone	External	Since you don't wish that participants in mesh network to use internal services of FreedomBox
IPv4 Addressing Method	Auto	Mesh networks usually have a DHCP server somewhere that provide your machine with IP configuration. If not, consult the operator and configure IP address setting accordingly with *Manual* method

Save the connection. Configure your machine for participation in the network by activating this connection. Currently, this connection has to be manually activated every time you need to join the network. In future, FreedomBox will do this automatically. You will now be able reach other nodes in the network. You will also be able to connect to the Internet via the mesh network if there is an Internet connection point somewhere in mesh as setup by the operators.

6.11.3.4.2 Creating a mesh network

To create your own mesh network and share your Internet connection with the rest of the nodes in the network:

1. Follow the instructions as provided above in step 1 of *Joining a mesh network* but choose and fix upon your own valid values for *SSID* (a name for you mesh network), *Frequency Band* (usually 2.4Ghz), *Channel* (1 to 11 in 2.4Ghz band) and *BSSID* (a hex value like 12:CA:DE:AD:BE:EF). Create this connection and activate it.
2. Follow the instructions as provided above in step 2 of *Joining a mesh network* but select *IPv4 Addressing Method* as *Shared*. This will provide automatic IP configuration to other nodes in the network as well as share the Internet connection on your machine (achieved using a second Wi-Fi interface, using Ethernet, etc.) with other nodes in the mesh network.

Spread the word about your mesh network to your neighbors and let them know the parameters you have provided when creating the network. When other nodes connect to this mesh network, they have to follow steps in *Joining a mesh network* but use the values for *SSID*, *Frequency Band* and *Channel* that you have chosen when you created the mesh network.

6.11.4 Manual Network Operation

FreedomBox automatically configures networks by default and provides a simplified interface to customize the configuration to specific needs. In most cases, manual operation is not necessary. The following steps describe how to manually operate network configuration in the event that a user finds FreedomBox interface to insufficient for task at hand or to diagnose a problem that FreedomBox does not identify.

On the command line interface:

For text based user interface for configuring network connections:

```
nmtui
```

To see the list of available network devices:

```
nmcli device
```

To see the list of configured connections:

```
nmcli connection
```

To see the current status of a connection:

```
nmcli connection show '<connection_name>'
```

To see the current firewall zone assigned to a network interface:

```
nmcli connection show '<connection_name>' | grep zone
```

or

```
firewall-cmd --zone=internal --list-all
firewall-cmd --zone=external --list-all
```

To create a new network connection:

```
nmcli con add con-name "<connection_name>" ifname "<interface>" type ethernet
nmcli con modify "<connection_name>" connection.autoconnect TRUE
nmcli con modify "<connection_name>" connection.zone internal
```

To change the firewall zone for a connection:

```
nmcli con modify "<connection_name>" connection.zone "<internal|external>"
```

For more information on how to use `nmcli` command, see its man page. Also for a full list of configuration settings and type of connections accepted by Network Manager see:

https://developer.gnome.org/NetworkManager/stable/ref-settings.html

To see the current status of the firewall and manually operate it, see the Firewall section.

6.12 Power

Power provides an easy way to restart or shut down FreedomBox. After you select "Restart" or "Shut Down", you will be asked to confirm.

"Restart" and "Shut Down" options can also be reached from the user dropdown menu on the top right.

6.13 Public Visibility (PageKite)

6.13.1 What is PageKite?

PageKite makes local websites and services publicly accessible immediately without creating yourself a public IP address. It does this by tunneling protocols such as HTTPS or SSH through firewalls and NAT. Using PageKite requires an account on a PageKite relay service. One such service is `https://pagekite.net`.

A PageKite relay service will allow you to create kites. Kites are similar to domain names, but with different advantages and drawbacks. A kite can have a number of configured services. PageKite is known to work with HTTP, HTTPS, and SSH, and may work with some other services, but not all.

6.13.2 Using PageKite

1. Create an account on a PageKite relay service.
2. Add a kite to your account. Note your kite name and kite secret.
3. In Plinth, go to the "Configure PageKite" tab on the Public Visibility (PageKite) page.
4. Check the "Enable PageKite" box, then enter your kite name and kite secret. Click "Save settings".
5. On the "Standard Services" tab, you can enable HTTP and HTTPS (recommended) and SSH (optional).
   - HTTP is needed to obtain the Let's Encrypt certificate. You can disable it later.
6. On the Certificates (Let's Encrypt) page, you can obtain a Let's Encrypt certificate for your kite name.

6.14 Secure Shell

6.14.1 What is Secure Shell?

FreedomBox runs `openssh-server` server by default allowing remote logins from all interfaces. If your hardware device is connected to a monitor and a keyboard, you may login directly as well. Regular operation of FreedomBox does not require you to use the shell. However, some tasks or identifying a problem may require you to login to a shell.

6.14.2 Setting Up A User Account

6.14.2.1 Plinth First Log In: Admin Account

When creating an account in FreedomBox's web interface for the first time, this user will automatically have administrator capabilities. `Admin` users are able to log in using ssh (see Logging In below) and have superuser privileges via sudo.

6.14.2.2 Default User Account

- Note: If you can access FreedomBox's web interface, then you don't need to do this. You can use the user account created in FreedomBox's web interface to connect to SSH.

The pre-built FreedomBox images have a default user account called "fbx". However the password is not set for this account, so it will not be possible to log in with this account by default.

There is a script included in the freedom-maker program, that will allow you to set the password for this account, if it is needed. To set a password for the "fbx" user:

1. Decompress the image file.
2. Get a copy of freedom-maker from `https://salsa.debian.org/freedombox-team/freedom-maker/`.
3. Run `sudo ./bin/passwd-in-image <image-file> fbx`.
4. Copy the image file to SD card and boot device as normal.

The "fbx" user also has superuser privileges via sudo.

6.14.3 Logging In

6.14.3.1 Local

To login via SSH, to your FreedomBox:

```
$ ssh fbx@freedombox
```

Replace `fbx` with the name of the user you wish to login as. `freedombox` should be replaced with the hostname or IP address of you FreedomBox device as found in the Quick Start process.

`fbx` is the default user present on FreedomBox with superuser privileges. Any other user created using Plinth and belonging to the group `admin` will be able to login. The `root` account has no password set and will not be able to login. Access will be denied to all other users.

`fbx` and users in `admin` group will also be able to login on the terminal directly. Other users will be denied access.

If you repeatedly try to login as a user and fail, you will be blocked from logging in for some time. This is due to `libpam-abl` package that FreedomBox installs by default. To control this behavior consult `libpam-abl` documentation.

6.14.3.2 SSH over Tor

If in Plinth you have enabled onion services via Tor, you can access your FreedomBox using ssh over Tor. On a GNU/Linux computer, install netcat-openbsd.

```
$ sudo apt-get install netcat-openbsd
```

Edit ~/.ssh/config to enable connections over Tor.

```
$ nano ~/.ssh/config
```

Add the following:

```
Host *.onion
  user USERNAME
  port 22
  ProxyCommand nc -X 5 -x 127.0.0.1:9050 %h %p
```

Replace USERNAME with, e.g., an `admin` username (see above).

Note that in some cases you may need to replace 9050 with 9150.

Now to connect to the FreedomBox, open a terminal and type:

```
$ ssh USERNAME@ADDRESS.onion
```

Replace USERNAME with, e.g., an `admin` username, and ADDRESS with the onion service address for your FreedomBox.

6.14.4 Becoming Superuser

After logging in, if you want to become the superuser for performing administrative activities:

```
$ sudo su
```

Make a habit of logging in as root *only when you need to*. If you aren't logged in as root, you can't accidentally break everything.

6.14.5 Changing Password

To change the password of a user managed by FreedomBox's web interface, use the change password page. However, the `fbx` default user is not managed by FreedomBox's web interface and its password cannot be changed through it.

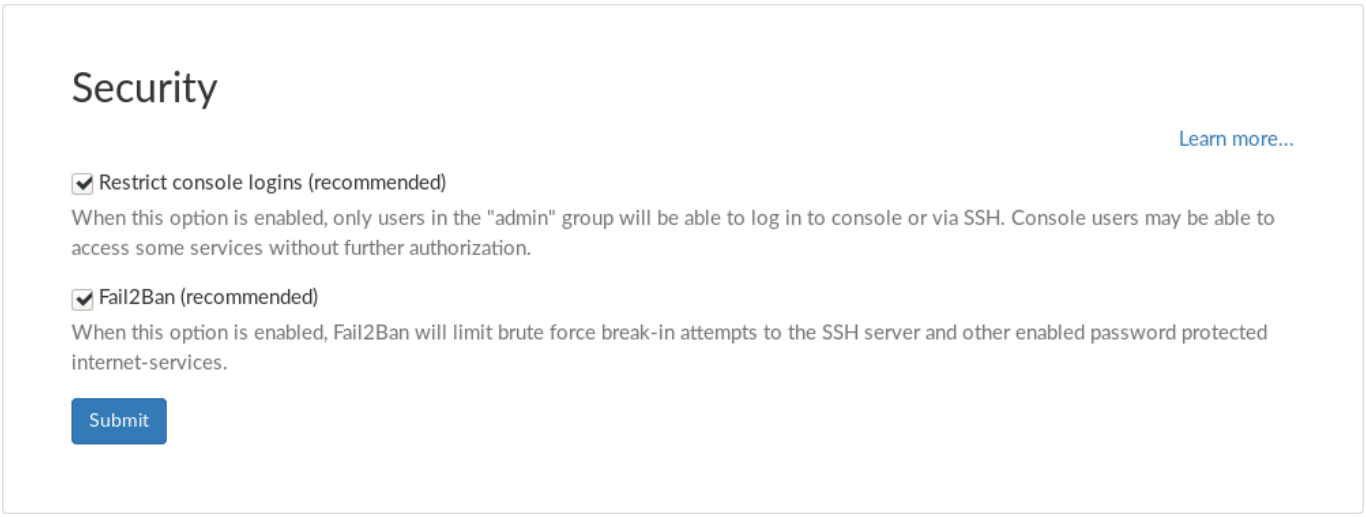
To change password on the terminal, log in to your FreedomBox as the user whose password you want to change. Then, run the following command:

```
$ passwd
```

This will ask you for your current password before giving you the opportunity to set a new one.

6.15 Security

When the *Restrict console logins* option is enabled, only users in the *admin* group will be able to log in via console, secure shell (SSH) or graphical login. When this option is disabled, any user with an account on FreedomBox will be able to log in. They may be able to access some services without further authorization. This option should only be disabled if all the users of the system are well trusted. If you wish to use your FreedomBox machine also as a desktop and allow non-admin users to login via GUI, this option must be disabled. You can define the list of users belonging to *admin* group in the Users section.



6.16 Service Discovery

Service discovery allows other devices on the network to discover your FreedomBox and services running on it. If a client on the local network supports mDNS, it can find your FreedomBox at <hostname>.local (for example: freedombox.local).

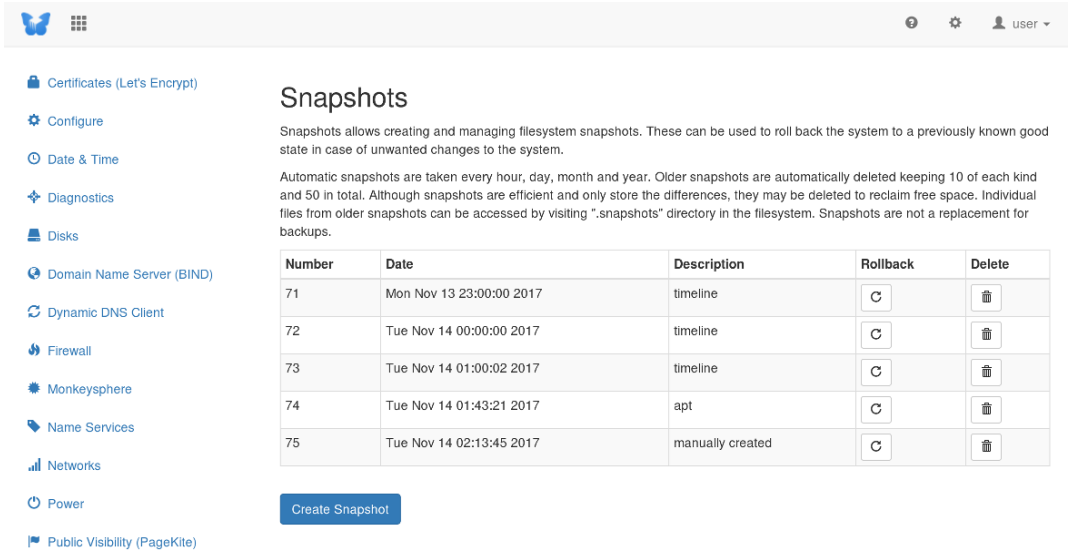
It also allows FreedomBox to discover other devices and services running on your local network.

Service discovery is not essential and works only on internal networks. It may be disabled to improve security especially when connecting to a hostile local network.

6.17 Snapshots

Snapshots allows you to create filesystem snapshots, and rollback the system to a previous snapshot.

- Note: This feature requires a Btrfs filesystem. All of the FreedomBox stable disk images use Btrfs.

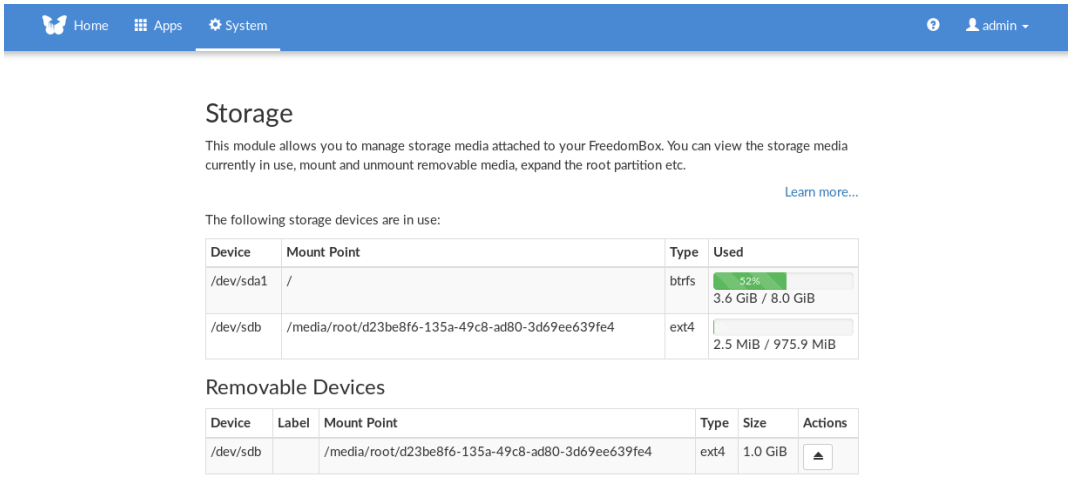


6.18 Storage

Storage allows you to see the storage devices attached to your FreedomBox and their disk space usage.

FreedomBox can automatically detect and mount removable media like USB flash drives. They are listed under the *Removable Devices* section along with an option to eject them.

If there is some free space left after the root partition, the option to expand the root partition is also available. This is typically not shown, since expanding the root partition happens automatically when the FreedomBox starts up for the first time.

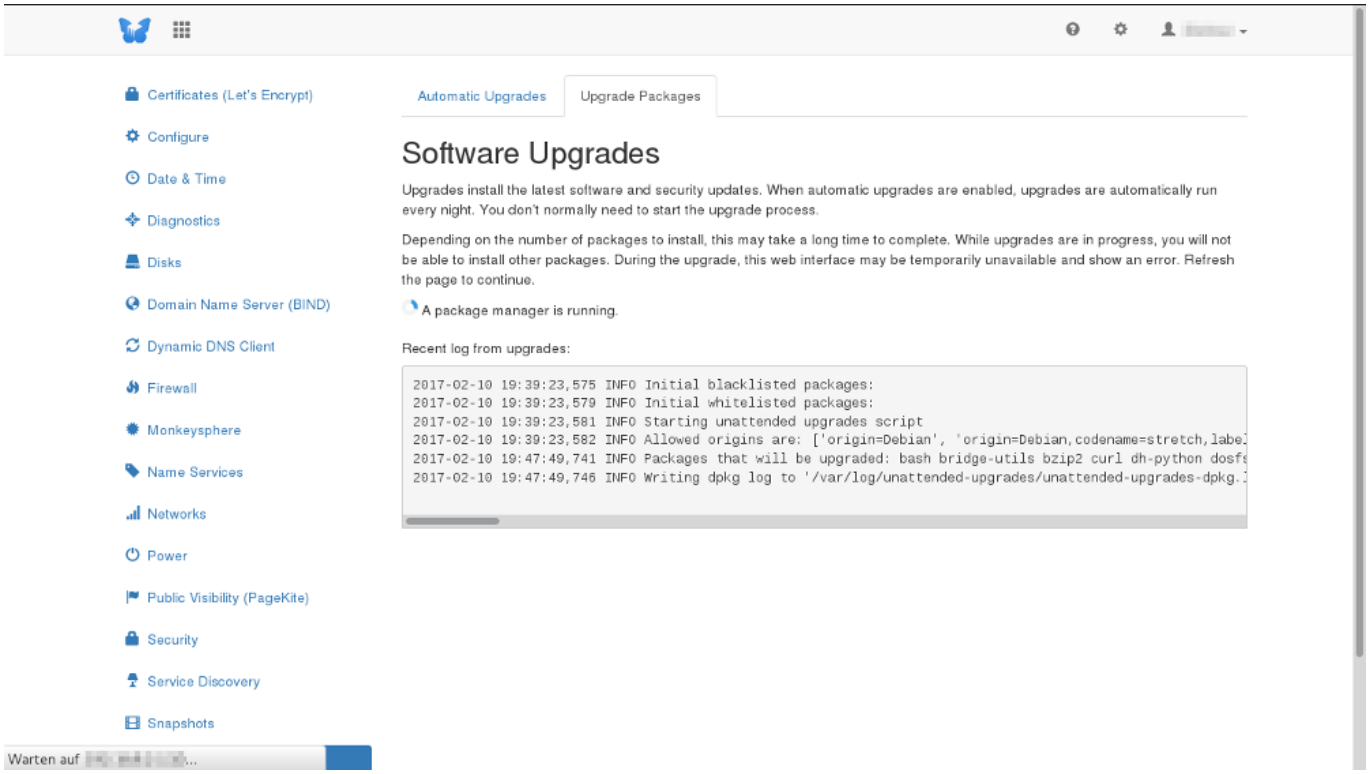


6.19 Software Updates

FreedomBox can automatically install security updates. On the *Update* page of the *System* section in FreedomBox web interface you can turn on automatic updates. This feature is enabled by default and there is no manual action necessary. It is strongly recommended that you have this option enabled to keep your FreedomBox secure.

Updates are performed every day at night. If you wish to shutdown FreedomBox every day after use, keep it running at night once a week or so to let the automatic updates happen. Alternatively, you can perform manual updates as described below.

Note that once the updates start, it may take a long time to complete. During automatic update process that runs every night or during manual update process, you will not be able to install apps from FreedomBox web interface.



6.19.1 When Will I Get the Latest Features?

Although updates are done every day for security reasons, latest features of FreedomBox will not propagate to all the users. The following information should help you understand how new features become available to users.

Stable Users: This category of users include users who bought the FreedomBox Pinoeer Edition, installed FreedomBox on a Debian stable distribution or users who downloaded the *stable* images from freedombox.org. As a general rule, only security updates to various packages are provided to these users. One exception to this rule is where FreedomBox service itself is updated when a release gains high confidence from developers. This means that latest FreedomBox features may become available to these users although not as quickly or frequently as *testing* users. If an app is available only in *testing* distribution but not in *stable* distribution, then that app will show up in the web interface but will not be installable by *stable* users. Some apps are also provided an exception to the rule of "security updates only" when the app is severely broken otherwise. Every two years, a major release of Debian stable happens with the latest versions of all the software packages and FreedomBox developers will attempt to upgrade these users to the new release without requiring manual intervention.

Testing Users: This category of users include users who installed FreedomBox on a Debian*testing* distribution or users who downloaded the *testing* images from freedombox.org. Users who use Debian *testing* are likely to face occasional disruption in the services and may even need manual intervention to fix the issue. As a general rule, these users receive all the latest features and security updates to all the installed packages. Every two weeks, a new version of FreedomBox is released with all the latest features and fixes. These releases will reach *testing* users approximately 2-3 days after the release.

Unstable Users: This category of users include users who installed FreedomBox on a Debian*unstable* distribution or users who downloaded the *unstable* images from freedombox.org. Users who use Debian *unstable* are likely to face occasional disruption

in the services and may even need manual intervention to fix the issue. As a general rule, these users receive all the latest features to all the installed packages. Every two weeks, a new version of FreedomBox is released with all the latest features and fixes. Theses releases will reach *unstable* users on the day of the release. Only developers, testers and other contributors to the FreedomBox project should use the *unstable* distribution and end users and advised against using it.

6.19.2 Manual Updates from Web Interface

To get updates immediately and not wait until the end of the day, you may want to trigger updates manually. You can do this by pressing the *Update now* button in *Manual update* tab for *Update* page in *System* section. Note that this step is not necessary if you have enabled *Auto-updates* as every night this operation is performed automatically.

When installing apps you may receive an error message such as

```
Error installing packages: E: dpkg was interrupted, you must manually run 'dpkg --configure ↩
    -a' to correct the problem
```

This is typically caused by shutting down FreedomBox while it is installing apps, while performing daily updates or during some other operations. This situation can be rectified immediately by running manual update.

6.19.3 Manual Updates from Terminal

Some software packages may require manual interaction for updating due to questions related to configuration. In such cases, FreedomBox updates itself and brings in new knowledge necessary to update the package by answering configuration questions. After updating itself, FreedomBox acts on behalf of the user and updates the packages by answering the questions. Until FreedomBox has a chance to update the package, such packages should not be be updated manually. The manual update triggered from the web interface is already mindful of such packages and does not update them.

In some rare situations, FreedomBox itself might fail to update or the update mechanism might fall into a situation that might need manual intervention from a terminal. To perform manual upgrades on the terminal, login into FreedomBox on a terminal (if you have monitor and keyboard connected), via a web terminal (using FreedomBox/Manual/Cockpit) or using a remote secure shell (see Secure Shell section). Then run the following commands:

```
$ sudo su -
Password: <enter user password here>
# dpkg --configure -a
# apt update
# apt -f install
# unattended-upgrade --debug
# apt install freedombox
# apt update
```

If `apt-get update` asks for a confirmation to change *Codename* or other release information, confirm *yes*. If during update of *freedombox* package, if a question about overwriting configuration files is asked, answer to install new configuration files from the latest version of the package. This process will upgrade only packages that don't require configuration file questions (except for *freedombox* package). After this, let FreedomBox handle the upgrade of remaining packages. Be patient while new releases of FreedomBox are made to handle packages that require manual intervention.

If you want to go beyond the recommendation to upgrade all the packages on your FreedomBox and if you are really sure about handling the configuration changes for packages yourself, run the following command:

```
$ apt dist-upgrade
```

6.20 Users and Groups

You can grant access to your FreedomBox for other users. Provide the Username with a password and assign a group to it. Currently the groups

- admin

- bit-torrent
- ed2k
- feed-reader
- syncthing
- web-search
- wiki

are supported.

The user will be able to log in to services that support single sign-on through LDAP, if they are in the appropriate group.

Users in the admin group will be able to log in to all services. They can also log in to the system through SSH and have administrative privileges (sudo).

A user's groups can also be changed later-on.

It is also possible to set an SSH public key which will allow this user to securely log in to the system without using a password. You may enter multiple keys, one on each line. Blank lines and lines starting with # will be ignored.

A user's account can be deactivated, which will temporarily disable the account.

6.20.1 Known Issues

- In Debian Stretch, the FreedomBox web interface does not distinguish between users and administrators. Every user added will have full access to the web interface.
  - This issue is fixed in Debian Buster and later.

7 Hardware

FreedomBox is designed to be the software for a consumer electronics device that is easy to setup, maintain and use. The project does not aim to create a custom hardware device ourselves, but instead we intend to partner with hardware vendors to build FreedomBox devices and also support existing hardware.

In addition to supporting various single board computers and other devices, FreedomBox also supports being installed in a virtual machine. Also, any Debian machine can be turned into a FreedomBox by installing the `freedombox` package. See the manual page for installing on Debian for more details.

7.1 Recommended Hardware

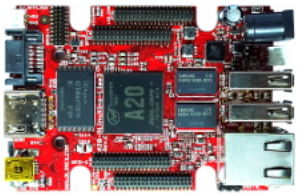
On April 22nd, 2019, the FreedomBox Foundation announced the sales of the Pioneer Edition FreedomBox Home Server Kits. This is the recommended pre-installed hardware for all users who don't wish to build their own FreedomBox by choosing the right components, downloading the image and preparing an SD card with FreedomBox.

The kit includes all the hardware needed for launching a FreedomBox home server on an Olimex A20-OLinuXino-LIME2 board. This product provides the perfect combination of open source hardware and free and open source software. By purchasing this product, you also support the FreedomBox Foundation's efforts to create and promote its free and open source server software.

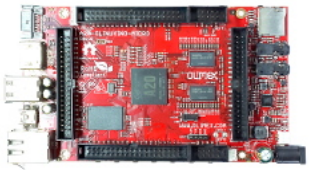
Pioneer Edition FreedomBox Home Server Kits

7.2 Supported Hardware

Use these hardware if you are able to download FreedomBox images and prepare an SD card by following the manual. If you wish for simper setup process, please buy the FreedomBox kits from recommended hardware instead. If you are using a board that uses SD cards, when you flash the FreedomBox image onto your SD card, we recommend that you use an SD card with at least 8GB of storage space.



A20 OLinuXino Lime2

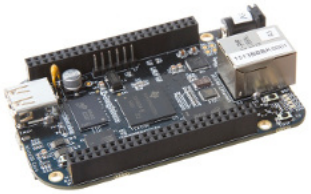


A20 OLinuXino MICRO

PC Engines APU

Cubietruck

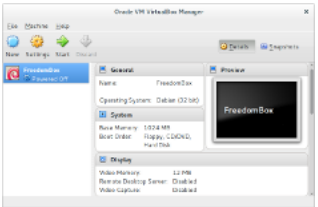
Cubieboard2



BeagleBone Black

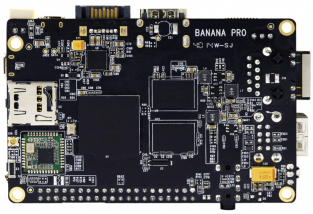
pcDuino3

Debian

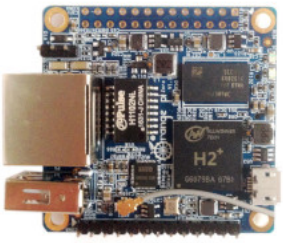


VirtualBox

Pine A64+



Banana Pro



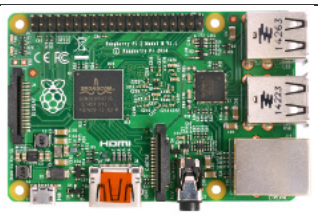
Orange Pi Zero

7.2.1 Hardware Comparison

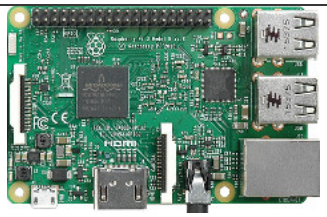
Name	Speed (GHz)	Debian arch	Ram (GB)	disk (GB)	battery	SATA	Ethernet speed	OSHW
APU.1D	1x2	amd64	2	-	-	✓	1000x3	✗
APU.1D4	1x2	amd64	4	-	-	✓	1000x3	✗
BeagleBone Black C	1	armhf/omap	½	4	-	-	100	✓
Cubieboard2	1x2	armhf/sunxi	1	4	✓	✓	100	✗
Cubieboard2-Dual	1x2	armhf/sunxi	1	-	✓	✓	100	✗
Cubieboard3/Cubietruck	1x2	armhf/sunxi	2	8	✓	✓	1000	✗
OLinuXino A20 LIME	1x2	armhf/sunxi	½	-	✓	✓	100	✓
OLinuXino A20 LIME2	1x2	armhf/sunxi	1	-	✓	✓	1000	✓
OLinuXino A20 MICRO	1x2	armhf/sunxi	1	-	✓	✓	100	✓
pcDunino3	1x2	armhf/sunxi	1	4	✓	✓	100	✗
Pine A64+	1.2x4	arm64/sunxi	½,1,2	-	-	-	1000	✗
Banana Pro	1.2x2	armhf/sunxi	1	-	-	✓	1000	✗
Orange Pi Zero	?x4	armhf/sunxi	¼,½	-	-	-	100	✗

7.3 Also Working Hardware

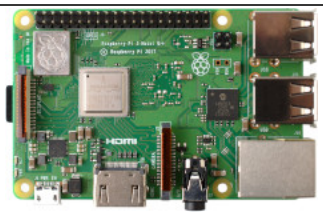
This hardware works but is not recommended because the hardware can't run entirely on free software:



Raspberry Pi 2



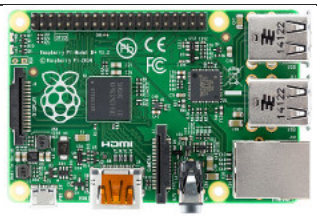
Raspberry Pi 3 Model B



Raspberry Pi 3 Model B+

7.4 Deprecated Hardware

This hardware was supported earlier but is no longer supported. If you downloaded an earlier image and are running FreedomBox on one of these hardware, you will keep getting software updates. However, no new images will be provided for these hardware. It is recommended that you migrate to newer, supported hardware using backup and restore.

 DreamPlug	 Raspberry Pi

Note: As FreedomBox is currently in the development stage, *Supported Hardware* means that FreedomBox images are built for said hardware and at least one developer has reported the basic functions to be working.

7.5 Adding Hardware Support

Although the project may focus on supporting specific devices, we are looking to support as much hardware as possible given that it is suitable for FreedomBox's needs. Take a look at the list of targeted hardware for more information.

If you are a developer, consider adding hardware support for your device by modifying Freedom Maker. If you have access to one of these targeted hardware devices and would like to work with us to make it run FreedomBox, please contact us!

7.6 Pioneer Edition FreedomBox

Pioneer FreedomBox Home Servers are produced and sold by Olimex, a company which specializes in Open Source Hardware. The kit includes pocket-sized server hardware, an SD card with the operating system pre-installed, and a backup battery which can power the hardware for 4-5 hours in case of outages. It sells for 82 euro. By purchasing this product, you also support the FreedomBox Foundation's efforts to create and promote its free and open source server software.

7.6.1 Product Features

The Pioneer Edition FreedomBox Home Server Kit includes all the hardware needed for launching a FreedomBox home server on an Olimex A20-OLinuXino-LIME2 board:

- the A20-OlinuXino-LIME2,
- a custom metal case with a laser-engraved FreedomBox logo,
- a high-speed 32GB micro SD card with the FreedomBox software pre-installed,

- a backup battery,
- a power adapter, and
- an Ethernet cable.

7.6.2 Recommended Hardware

This is the hardware recommended for all users who just want a turn-key FreedomBox, and **don't** want to **build** their own one.

(Building your own FreedomBox means some technical stuff like choosing and buying the right components, downloading the image and preparing the SD card).

7.6.3 Availability

The Pioneer Edition FreedomBox Home Server is the first commercially available version of FreedomBox.

- Price: 82 EUR
- Olimex Store

7.6.4 Hardware Specifications

- Open Source Hardware (OSHW): Yes
- CPU: Allwinner A20, ARM Cortex-A7 @ 1GHz dual-core
- RAM: 1 GiB DDR3
- Storage: 32GB class 10+ microSD card pre-loaded with FreedomBox
- Battery: 3.3V Li-Po, 1400mAh (4-5 hours of backup without additional devices connected via USB)
- Ethernet: 10/100/1000, RJ45 (1 meter cable included)
- Power adapter: 110-220 V input, 5V output, EU style (with optional UK or US sockets)
- Box: Custom metallic box with FreedomBox decal

The kits run entirely on Free Software. They work with Kernel and u-boot from Debian repositories. Even the boot firmware in ROM called BROM is free software (GPLV2+).

7.6.5 Download

The kits come with an SD card pre-loaded with FreedomBox. There's **NO need to download images**.

However, if you wish to reset your devices to a pristine state, then you can do so with the the image provided. Follow the instructions on the download page to create a FreedomBox SD card and boot the device. Make sure to download the Pioneer Edition images. These SD card images are meant for use with the on-board SD card slot and won't work when used with a separate SD card reader connected via USB.

An alternative to downloading these images is to install Debian on the device and then install FreedomBox on it.

7.6.6 Build Image

FreedomBox images for this hardware can be built using Freedom Maker.

7.6.7 Known Issues

- The image that shipped with the kits uses a slightly modified u-boot from Debian and not stock Debian like the rest of FreedomBox. So, if you wish to get the source code, please use the FreedomBox team's u-boot repository.

7.6.8 Obtaining Source Code

After you purchase and receive your Pioneer Edition FreedomBox, you may want to obtain the source code of the software running in it. Continue reading this section for instructions.

FreedomBox is fully free software and you can obtain the source code to study, modify and distribute improvements.

7.6.8.1 From within FreedomBox

FreedomBox is made up of several software programs and you can obtain the source code to any of them. These instructions are similar to obtaining and building source code for Debian since FreedomBox is a pure blend of Debian. Using this process you can obtain the source code to the exact version of the package you are currently using in FreedomBox.

1. To see the list of software packages installed on your FreedomBox, run the following in a terminal:

   ```
   dpkg -l
   ```

2. To obtain the source code for any of those programs, then run:

   ```
   apt source <package_name>
   ```

   This requires that the file /etc/apt/sources.list file contains the information about the source code repositories. These are present by default on all FreedomBox images. If you have installed FreedomBox using a package from Debian, you need to ensure that source repositories are added in the file.

3. To build the package from source code, first install its dependencies

   ```
   apt build-dep <package_name>
   ```

   Switch to the source directory created by the *apt source* command:

   ```
   cd <source_directory>
   ```

   Then build the package

   ```
   dpkg-buildpackage -rfakeroot -uc
   ```

4. Install the package:

   ```
   dpkg -i ../<built_package>.deb
   ```

7.6.8.2 Other Ways to Obtain Source Code

1. Source code for any of the packages can be browsed and searched using the web interface at sources.debian.org. For example, see the plinth package.

2. Source code and pre-built binary package for any version of a package including historic versions can be obtained from snapshot.debian.org. For example, see the plinth package.

3. You can also obtain the links to upstream project homepage, upstream version control, Debian's version control, changelog, etc. from the Debian tracker page for a project at tracker.debian.org. For example, see the tracker page for plinth package.

4. You can build and install a package from its Debian's version control repository. For example,

```
git clone https://salsa.debian.org/freedombox-team/plinth
cd plinth
apt build-dep .
dpkg-buildpackage -rfakeroot -uc
dpkg -i ../plinth*.deb
```

7.6.8.3 Building Disk Images

You can also build FreedomBox disk images for various hardware platforms using the freedom-maker tool. This is also available as a Debian package and source code for it may be obtained using the above methods. Build instructions for creating disk images are available as part of the source code for freedom-maker package.

FreedomBox disk images are built and uploaded to official servers using automated Continuous Integration infrastructure. This infrastructure is available as source code too and provides accurate information on how FreedomBox images are built.

7.6.8.4 U-boot on Pioneer Edition Images

There is one minor exception to the u-boot package present on the hardware sold as FreedomBox Home Server Kits Pioneer Edition. It contains an small but important fix that is not part of Debian sources. The fork of the Debian u-boot source repository along with the minor change done by the FreedomBox is available as a separate repository. We except this change to be available in upstream u-boot eventually and this repository will not be needed. This package can be built on a Debian armhf machine as follows (cross compiling is also possible, simply follow instructions for cross compiling Debian packages):

```
apt install git git-buildpackage
git clone https://salsa.debian.org/freedombox-team/u-boot.git
cd u-boot
pbuilder create --distribution=buster
gbp buildpackage --git-pbuilder
```

The u-boot Debian package will be available in *u-boot-sunxi*.deb*. This package will contain

```
mkdir temp
dpkg -x u-boot-suxi*.deb temp
unxz <lime2_image_built_with_freedom_maker>
dd if=temp/usr/lib/u-boot/A20-OLinuXino-Lime2/u-boot-sunxi-with-spl.bin of=<lime2.img> seek ↩
    =8 bs=1k conv=notrunc
```

The resulting image will have the modified u-boot in it.

7.7 Cubietruck

7.7.1 FreedomBox Danube Edition

FreedomBox Danube Edition is a custom casing around Cubietruck and an SSD-hard drive.

7.7.2 Cubietruck / Cubieboard3

Cubietruck (Cubieboard3) is a single board computer with very good performance compared to many other boards. FreedomBox images are built for this device. To use this board as FreedomBox, a separate USB WiFi device that does not require non-free firmware is recommended.

Recommendation: If you are using a board that uses SD cards, when you flash the FreedomBox image onto your SD card, we recommend that you use an SD card with at least 8GB of storage space.

7.7.3 Download

FreedomBox SD card images are provided for this hardware. These SD card images are meant for use with the on-board SD card slot and do not work when used with a separate SD card reader connected via USB.

An alternative to downloading these images is to install Debian on the Cubietruck and then install FreedomBox on it.

7.7.4 Build Image

FreedomBox images for this hardware can be built using Freedom Maker.

7.7.5 Availability

FreedomBox Danube Edition

- A limited number of units are planned to be shipped along with the release of FreedomBox. If you wish to get one, express your interest.

Cubietruck / Cubieboard3

- Price: 89 USD
- List of suppliers

7.7.6 Hardware

- Open Hardware: No
- CPU: Allwinner A20, ARM Cortex-A7 @ 1GHz dual-core
- RAM: 2 GiB DDR3 @ 480 MHz
- Storage: 8 GB NAND flash built-in, 1x microSD slot
- Architecture: armhf
- Ethernet: 10/100/1000, RJ45
- WiFi: Broadcom BCM4329/BCM40181 (no free WiFi drivers + firmware available)
- SATA: 1x 2.0 port

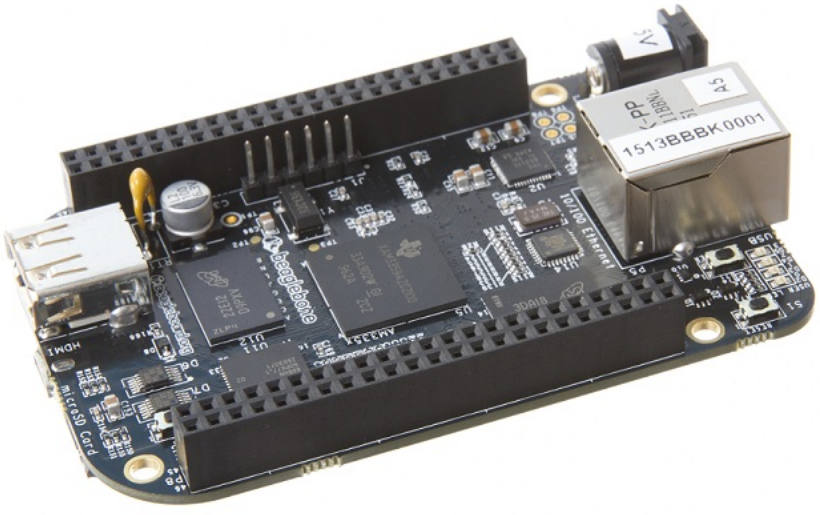
7.7.7 Non-Free Status

- Non-free blobs required: ?
- WiFi: no free WiFi drivers + firmware available
- Works with stock Debian kernel: yes

7.7.8 Known Issues

- The on-board WiFi does not work with free software. A separate USB WiFi device is recommended.

7.8 Beagle Bone Black



Beagle Bone Black (Revision C.1) is an Open Source Hardware (OSHW) single board computer. This means that the designer is actively helping people using the platform for their own designs, and supports them in adding hardware functionality and production advice. This is a part of freedom that is often overlooked, but very much aligned with the FreedomBox goals. FreedomBox images are built and tested for this device. To use this device as a FreedomBox, a separate USB WiFi device that does not require non-free firmware is recommended.

Recommendation: If you are using a board that uses SD cards, when you flash the FreedomBox image onto your SD card, we recommend that you use an SD card with at least 8GB of storage space.

7.8.1 Download

FreedomBox SD card images are available for this device. Follow the instructions on the download page to create a FreedomBox SD card and boot the device.

Note: This image is for BeagleBone Black (Revision C.1) only. It will not work on the BeagleBone Green, and also not on the Revisions A&B. If you have such a device and would like to help getting FreedomBox to run on it, contact us!

An alternative to downloading these images is to install Debian on the BeagleBone and then install FreedomBox on it.

7.8.2 Build Image

FreedomBox images for this hardware can be built using Freedom Maker.

7.8.3 Availability

- Price: ~ 59 USD (50 EUR)
- Mouser Electronics
- Full list of suppliers

7.8.4 Hardware

- Open Source Hardware (OSHW): Yes
- CPU: AM335x 1GHz ARM Cortex-A8
- RAM: 512MB DDR3L 800 Mhz
- Storage: Onboard 4GB, 8bit Embedded MMC and microSD
- Architecture: armhf
- Ethernet: 10/100, RJ45
- WiFi: None, use a USB WiFi device
- SATA: None

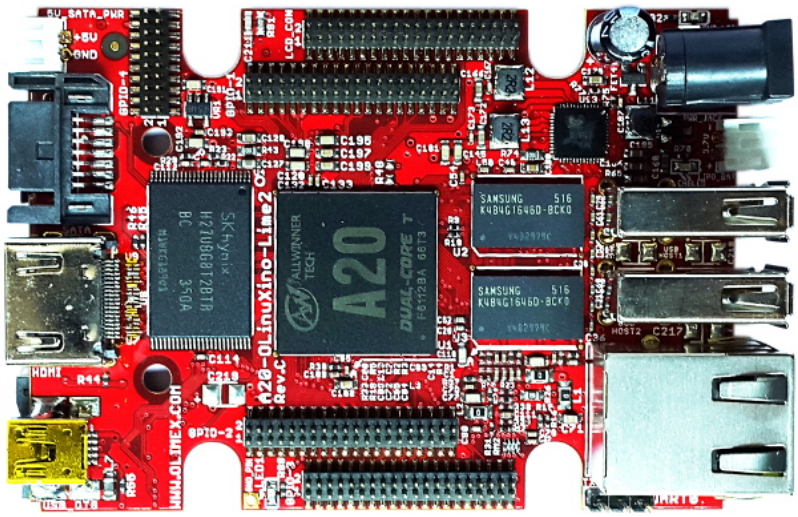
7.8.5 Non-Free Status

- Non-free blobs required: No
- WiFi: Not available
- Works with stock Debian kernel: Yes

7.8.6 Known Issues

None

7.9 A20 OLinuXino Lime2



Olimex's A20 OLinuXino Lime2 is a fully Open Source Hardware (OSHW) single board computer. This means that the designer is actively helping people using the platform for their own designs, and supports them in adding hardware functionality and production advice. This is a part of freedom that is often overlooked, but very much aligned with the FreedomBox goals. It uses the Allwinner A20 Dual Core ARM processor. FreedomBox images are built and tested for this device starting with version 0.7. To use this device as a FreedomBox, a separate USB WiFi device that does not require non-free firmware is recommended.

Recommendation: If you are using a board that uses SD cards, when you flash the FreedomBox image onto your SD card, we recommend that you use an SD card with at least 8GB of storage space.

7.9.1 Similar Hardware

The following similar hardware will also work well with FreedomBox.

- Olimex's A20 OLinuXino Lime2 4GB. This hardware merely has extra 4GB NAND storage that is not used by FreedomBox.

7.9.2 Download

FreedomBox SD card images are available for this device. Follow the instructions on the download page to create a FreedomBox SD card and boot the device. These SD card images are meant for use with the on-board SD card slot and won't work when used with a separate SD card reader connected via USB.

An alternative to downloading these images is to install Debian on the device and then install FreedomBox on it.

7.9.3 Build Image

FreedomBox images for this hardware can be built using Freedom Maker.

7.9.4 Availability

- Price: 45 EUR (A20 OLinuXino Lime2)
- Price: 55 EUR (A20 OLinuXino Lime2 4GB)
- Olimex Store

7.9.5 Hardware

- Open Source Hardware (OSHW): Yes
- CPU: Allwinner A20, ARM Cortex-A7 @ 1GHz dual-core
- RAM: 1 GiB DDR3
- Storage: 4 GB NAND flash built-in (only on 4GB model), 1x microSD slot
- Architecture: armhf
- Ethernet: 10/100/1000, RJ45
- WiFi: None, use a USB WiFi device
- SATA: 1x port

7.9.6 Non-Free Status

- Non-free blobs required: No
- WiFi: Not available
- Works with stock Debian kernel: Yes
- Boot Firmware: BROM (GPLV2+)

7.9.7 Known Issues

- Revision C hardware has poor performance when receiving Ethernet data in Gigabit mode. To workaround the problem, you can switch to 100 Mbps mode instead of Gigabit mode. Login to your FreedomBox as root (or plugin the SD card into another computer) and create the file /etc/NetworkManager/dispatcher.d/20-fix-ethernet-problem with the following contents:

```
set -e  # Exit with code on error

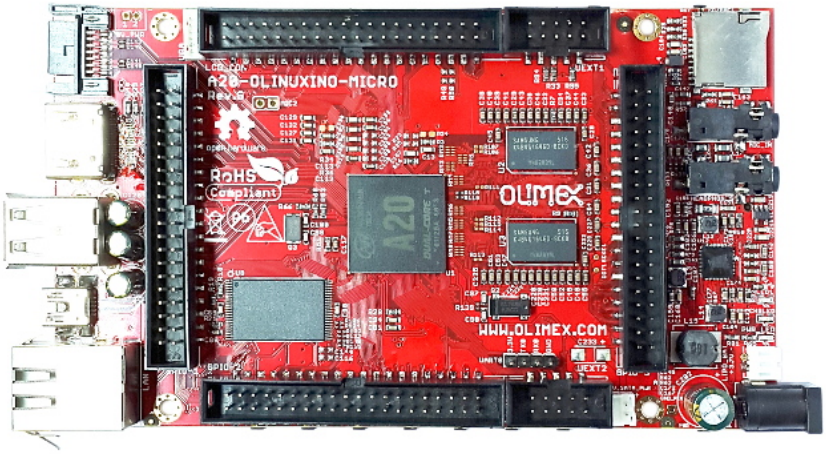
IFACE="$1"
ACTION="$2"

if [[ "$IFACE" != "eth0" ]]; then
    exit 0
fi

case ${ACTION} in
    up)
        logger "Setting up $IFACE in 100Mbps mode"
        mii-tool eth0 -A 100BaseTx-FD
        ;;
    *)
        ;;
esac
```

- Revision G2 hardware has poor performance when transmitting Ethernet data in Gigabit mode. Download and use the Pioneer Edition image to fix the issue. It contains a slightly modified u-boot. The above workaround to put the Ethernet into 100 Mbps mode also fixes this issue.
- Revision K hardware is not working properly.

7.10 A20 OLinuXino MICRO



Olimex's A20 OLinuXino MICRO is a fully Open Source Hardware (OSHW) single board computer. This means that the designer is actively helping people using the platform for their own designs, and supports them in adding hardware functionality and production advice. This is a part of freedom that is often overlooked, but very much aligned with the FreedomBox goals. It uses the Allwinner A20 Dual Core ARM processor. FreedomBox images are built and tested for this device starting with version 0.7. To use this device as a FreedomBox, a separate USB WiFi device that does not require non-free firmware is recommended.

Recommendation: If you are using a board that uses SD cards, when you flash the FreedomBox image onto your SD card, we recommend that you use an SD card with at least 8GB of storage space.

7.10.1 Similar Hardware

The following similar hardware will also work well with FreedomBox.

- Olimex's A20 OLinuXino MICRO 4GB. This hardware merely has extra 4GB NAND storage that is not used by FreedomBox.

7.10.2 Download

FreedomBox MicroSD card images are available for this device. Follow the instructions on the download page to create a FreedomBox MicroSD card and boot the device. These MicroSD card images are meant for use with the on-board MicroSD card slot and won't work on the SD card slot or when using a separate MicroSD card reader connected via USB.

An alternative to downloading these images is to install Debian on the device and then install FreedomBox on it.

7.10.3 Build Image

FreedomBox images for this hardware can be built using Freedom Maker.

7.10.4 Availability

- Price: 50 EUR (A20 OLinuXino MICRO)
- Price: 63 EUR (A20 OLinuXino MICRO 4GB)
- Olimex Store

7.10.5 Hardware

- Open Source Hardware (OSHW): Yes
- CPU: Allwinner A20, ARM Cortex-A7 @ 1GHz dual-core
- RAM: 1 GiB DDR3
- Storage: 4 GB NAND flash built-in (only on 4GB model), 1x microSD slot
- Architecture: armhf
- Ethernet: 10/100, RJ45
- WiFi: None, use a USB WiFi device
- SATA: 1x port

7.10.6 Non-Free Status

- Non-free blobs required: No
- WiFi: Not available
- Works with stock Debian kernel: Yes
- Boot Firmware: BROM (GPLV2+)

7.10.7 Known Issues

- Not visible on local network
- When booting the 'stable' image (made on 2017-06-18) the board does not automatically get an IP address from the router's DHCP server over ethernet. Booting the 'testing' image (2018-06) the board does get an IP address. Tested on MICRO hardware revision J. see also: https://www.olimex.com/forum/index.php?topic=5839.msg24167#msg24167

7.11 APU

PC Engines APU 1D is a single board computer with 3 Gigabit ethernet ports, a powerful AMD APU and Coreboot firmware. FreedomBox images built for AMD64 machines are tested to work well for it. For using this board as FreedomBox, a USB WiFi device that does not require non-free firmware is recommended.

Recommendation: If you are using a board that uses SD cards, when you flash the FreedomBox image onto your SD card, we recommend that you use an SD card with at least 8GB of storage space.

7.11.1 Similar Hardware

Although untested, the following similar hardware is also likely to work well with FreedomBox.

- Using amd64 image:
  - apu1c
  - apu1c4
  - apu1d4
  - apu2b2
  - apu2b4
  - apu2c0
  - apu2c2
  - apu2c4
  - apu3a2
  - apu3a4
  - apu3b2
  - apu3b4
- Using i386 image:
  - alix1d
  - alix1e
  - alix2d2
  - alix2d3
  - alix2d13
  - alix3d2
  - alix3d3
  - alix6f2

7.11.2 Download

FreedomBox disk images for this hardware are available. Follow the instructions on the download page to create a FreedomBox SD card, USB disk, SSD or hard drive and boot into FreedomBox. Pick the image meant for all amd64 machines.

An alternative to downloading these images is to install Debian on the APU and then install FreedomBox on it.

7.11.3 Networking

The first network port, the left most one in the above picture, is configured by FreedomBox to be an upstream Internet link and the remaining 2 ports are configured for local computers to connect to.

7.11.4 Build Image

FreedomBox images for this hardware, which is for all amd64 machines, can be built using Freedom Maker.

7.11.5 Availability

- Price: 110 - 170 USD (depending on the board and supplier)
- PC Engines
- Full list of suppliers

7.11.6 Hardware

- Open Hardware: No
- CPU: AMD G series T40E
- RAM: 2 GB DDR3-1066 DRAM
- Storage: SD card, External USB
- Architecture: amd64
- Ethernet: 3 Gigabit Ethernet ports
- WiFi: None, use a USB WiFi device
- SATA: 1 m-SATA and 1 SATA

7.11.7 Non-Free Status

- Non-free blobs required: No
- WiFi: Not available
- Works with stock Debian kernel: Yes
- Boot firmware: Coreboot

7.11.8 Known Issues

None

7.12 pcDuino3

LinkSprite pcDuino3S is a single board computer running on Allwinner A20 and sold with a good case. FreedomBox images are built and tested for this device for images built after June 2017. For using this board as FreedomBox, a USB WiFi device that does not require non-free firmware is recommended.

Note: The FreedomBox logo is simply a sticker on top of device brought from store.

Recommendation: If you are using a board that uses SD cards, when you flash the FreedomBox image onto your SD card, we recommend that you use an SD card with at least 8GB of storage space.

7.12.1 Similar Hardware

Although untested, the following similar hardware is also likely to work well with FreedomBox.

- http://www.linksprite.com/linksprite-pcduino3/pcDuino3
- http://www.linksprite.com/linksprite-pcduino3/pcDuino3B

7.12.2 Download

FreedomBox disk images for this hardware are available. Follow the instructions on the download page to create a FreedomBox SD card, USB disk, SSD or hard drive and boot into FreedomBox. Pick the image meant for pcduino3.

An alternative to downloading these images is to install Debian on the APU and then install FreedomBox on it.

7.12.3 Build Image

FreedomBox images for this hardware can be built using Freedom Maker.

7.12.4 Availability

- Price: 89 USD
- LinkSprite
- Full list of suppliers

7.12.5 Hardware

- Open Hardware: No
- CPU: AllWinner A20 SoC, 1GHz ARM Cortex A7 Dual Core
- RAM: 1 GB
- Storage: SD card, 4 GB onboard flash
- Architecture: armhf
- Ethernet: 10/100 Mbps
- WiFi: Built-in WiFi requires non-free firmware, use a USB WiFi device instead
- SATA: 1 SATA host socket

7.12.6 Non-Free Status

- Non-free blobs required: No
- WiFi: Requires non-free firmware
- Works with stock Debian kernel: Yes
- Boot Firmware: BROM (GPLV2+)

7.12.7 Known Issues

None

7.13 Pine A64+

Pine A64+ is an affordable single board computer with good performance.

Recommendation: Pine A64+ series of boards do not have built-in storage. If installing to a microSD card, it is recommended to choose a microSD card of class 10 or better with at least 8 GB of storage.

7.13.1 Similar Hardware

- Both 1GB and 2GB versions of Pine A64+ are supported with the same FreedomBox image.
- Pine A64-LTS is not supported yet.

7.13.2 Download

FreedomBox SD card images for this hardware are available. Follow the instructions on the download page to create a FreedomBox SD card and boot into FreedomBox. Pick the image meant for Pine A64+.

An alternative to downloading these images is to install Debian on the device and then install FreedomBox on it.

7.13.3 Build Image

FreedomBox images for this hardware can be built using freedom-maker.

7.13.4 Availability

- Price: 29 USD (for the 2 GB variant), 21 USD (for the 1 GB variant)
- Pine A64+ with 1 GB RAM at Pine64 Store
- Pine A64+ with 2 GB RAM at Pine64 Store

7.13.5 Hardware

- Open Source Hardware (OSHW): No
- CPU: Allwinner A64, Quad-core ARM Cortex A53 64-bit processor
- RAM: 3 variants - 512 MB (not recommended), 1 GB and 2 GB (recommended)
- Storage: SD card, eMMC (module sold separately but not tested with FreedomBox)
- Architecture: arm64
- Ethernet: Gigabit Ethernet port
- Battery: Supports battery backup using a Li-Po battery
- WiFi: None, use a USB WiFi device
- SATA: None

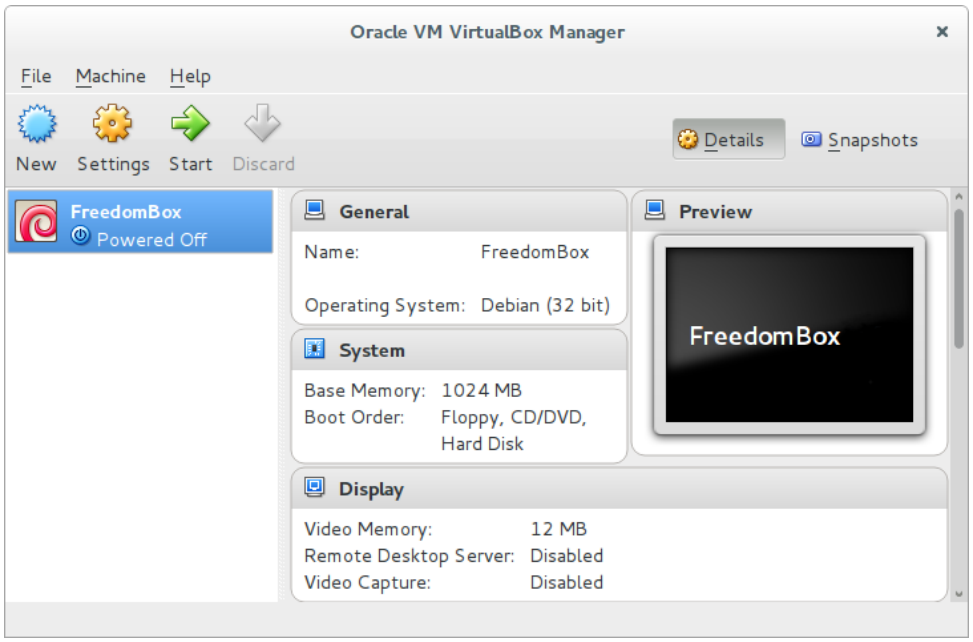
7.13.6 Non-Free Status

- Non-free blobs required: No
- WiFi: Not available
- Works with stock Debian kernel: Yes

7.13.7 Known Issues

None

7.14 VirtualBox



This page will help you get started with using FreedomBox on a virtual machine using VirtualBox. While VirtualBox images are primarily used for testing and development, they can also be used for regular use if you have spare resources on one of your machines. This setup is useful if:

- You don't own one of the supported hardware devices.
- You don't use Debian GNU/Linux as your operating system.
- You don't want to disturb your Debian installation to try out FreedomBox.

Prebuilt FreedomBox images for VirtualBox are routinely made available in VirtualBox's own VDI image file format. They contain a Debian GNU/Linux operating system and an installation of FreedomBox with all dependencies ready to run on any OS supported by VirtualBox (Windows, Linux, Macintosh, and Solaris).

A more adventurous alternative to downloading one of these images is to install Debian on VirtualBox and then install FreedomBox on it.

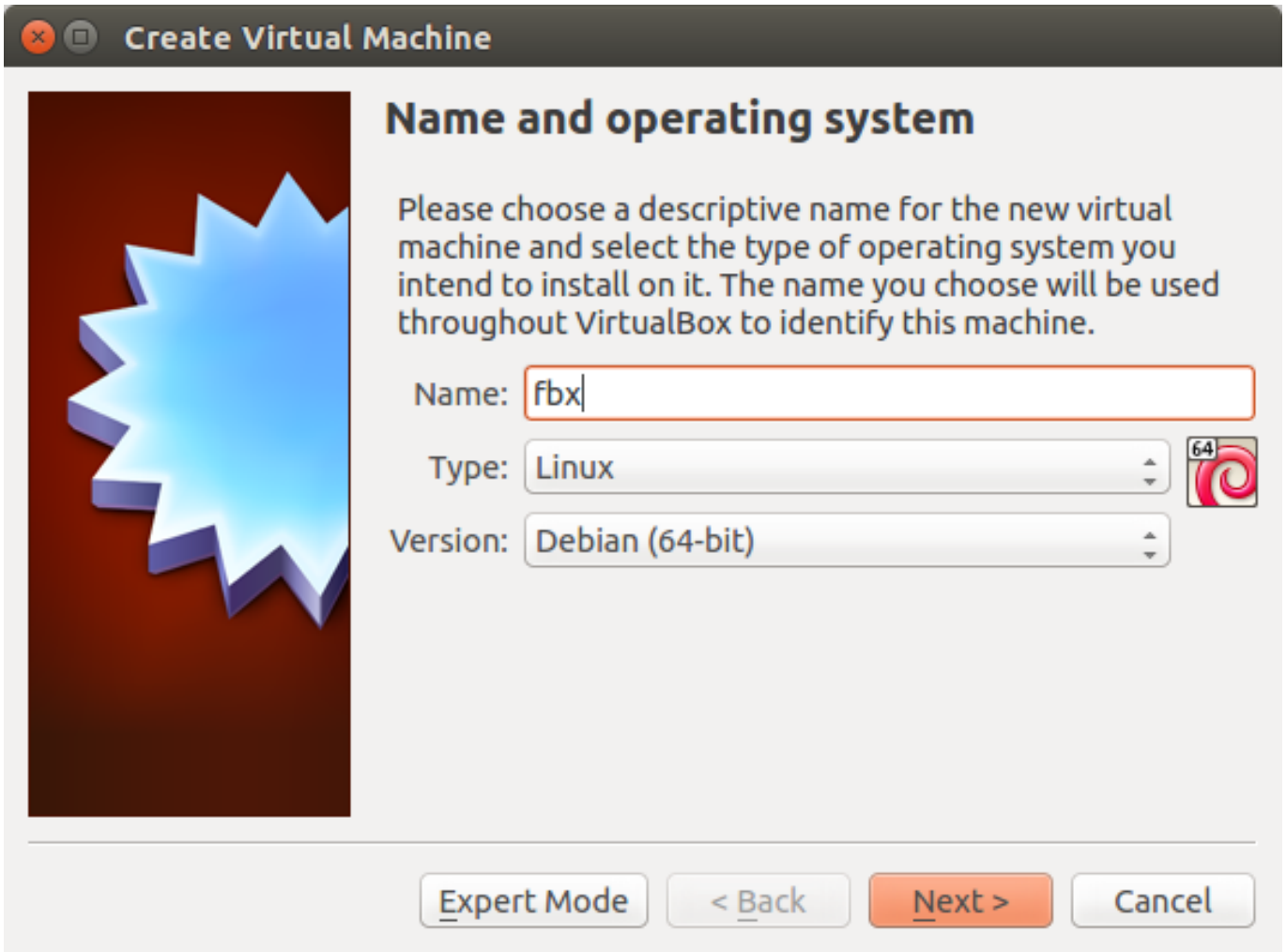
VirtualBox itself is available from `https://www.virtualbox.org/` (or your distribution's package manager).

7.14.1 Download

Follow the instructions on the download page to download and verify a VirtualBox image. The latest images are available on freedombox.org.

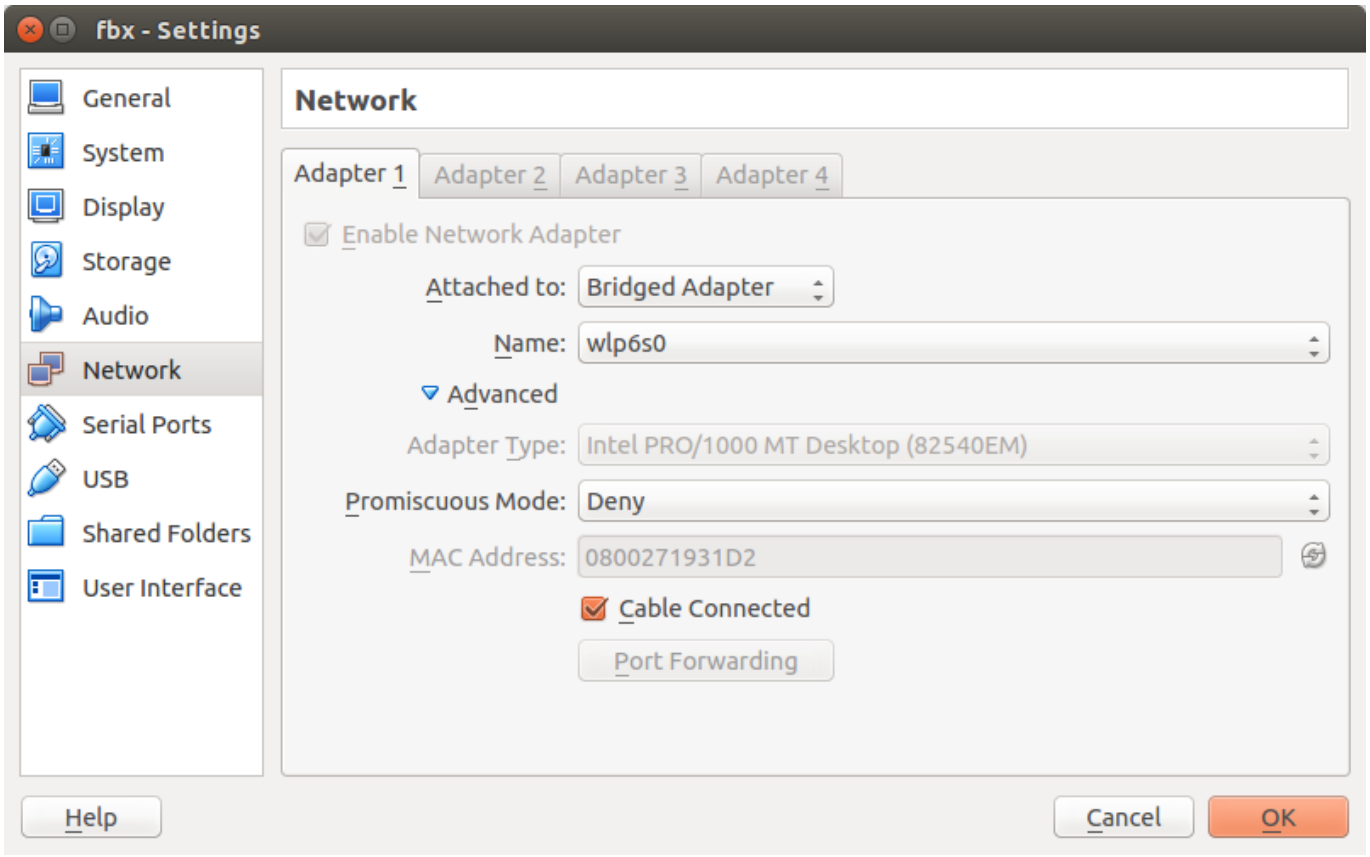
7.14.2 Creating a Virtual Machine

1. Decompress the downloaded VDI image (tool for Windows, Mac).
2. Create a new VM in the VirtualBox UI with OS type *Linux* and Version *Debian* (32/64-bit according to the downloaded image).



1. In the *Hard disk* dialog choose *Use an existing virtual hard disk file* and select the .vdi file you extracted in step 1.

1. When created, go to the virtual machine's Settings -> [Network] -> [Adapter 1]->[Attached to:] and choose the network type your want the machine to use according to the explanation in Network Configuration below. The recommended type is the *Bridged adapter* option, but be aware that this exposes the FreedomBox's services to your entire local network.

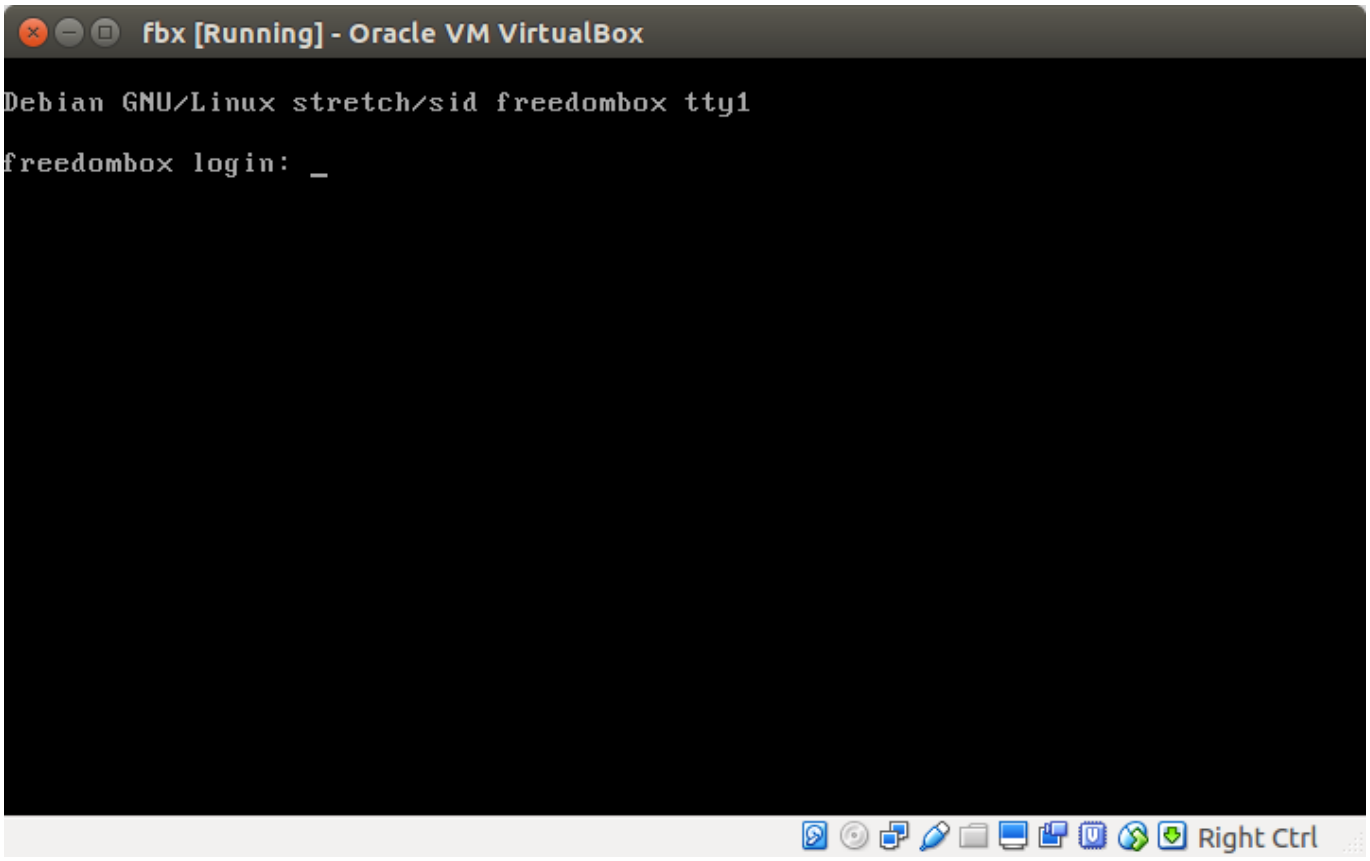


Note: It is important to make sure that you have provided the correct network interface in the above step. For example, if the virtual machine is running on a laptop connected to a Wi-Fi network, then the wireless interface (starts with *wlp*) must be chosen as shown in the screenshot.

7.14.3 First Boot

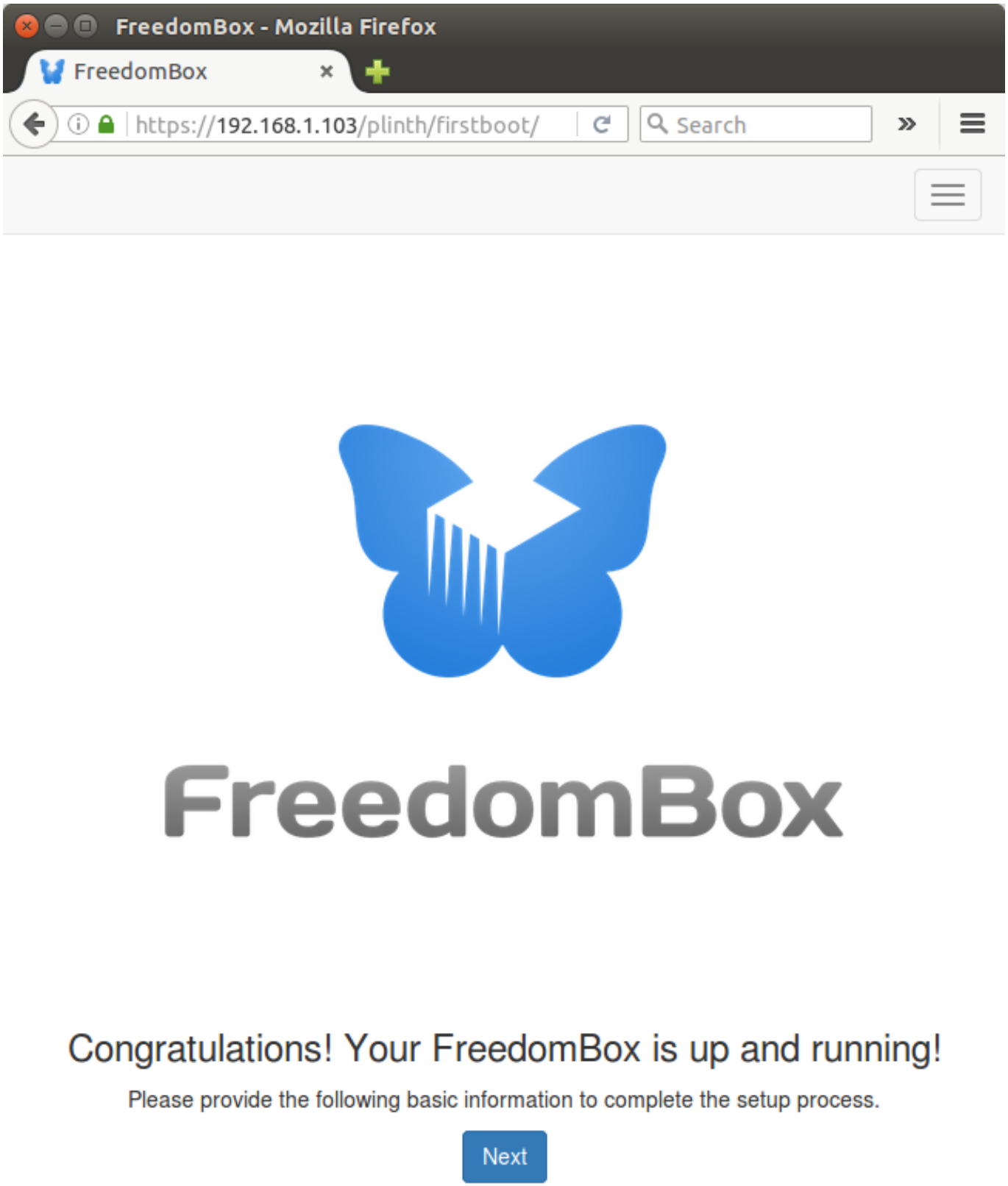
When satisfied with the VM settings click the start button in the VirtualBox UI and your new FreedomBox will boot.

The console of the VM will show the textual screen below when finished booting, from here most interaction with FreedomBox will be through the web interface (aka. Plinth) in a browser.



If everything went well so far, you should be able to access the web interface of FreedomBox by pointing a browser on the host machine to `https://freedombox.local`.

In case freedombox.local cannot be resolved, you need to find out your FreedomBox's IP address as described in Finding out the IP address of the virtual machine. Then access this IP from a web browser which is on the same network as the VM (f.ex. the host). If all is well, you are now presented with a welcome message and invited to complete the *first boot* process.



This mainly consist of creating an administrative user for the system.

7.14.4 Using

See the FreedomBox usage page for more details.

You can log in to the Debian GNU/Linux system as the user created during Plinth first boot on the VirtualBox console or remotely via ssh.

After logging in, you can become root with the command `sudo su`.

7.14.5 Build Image

If you wish to build your own images instead of downloading available images, it can be done using Freedom Maker.

7.14.6 Tips & Troubleshooting

7.14.6.1 Network Configuration

VirtualBox provides many types of networking options. Each has its advantages and disadvantages. For more information about how various networking types work in VirtualBox, see VirtualBox's networking documentation. https://www.virtualbox.org/manual/ch06.html

For a simple setup, it is recommended that you use a single network interface in your guest machine. This will make the first boot script automatically configure that interface as an `internal` network with `automatic` network configuration. Inside the guest machine, the networking is configured automatically and all the services are made available on this network interface. For more information on how networks are configured by default in FreedomBox, see Networks section.

What remains is to make those services available to the host machine or to other machines in the network. You must then choose one of the following types of networking for the network interface on your guest machine. To set a particular type of network for the guest's network adapter, go to the guest VM's settings then the network options and then select the adapter you wish to configure. There, set the network type from the available list of networks.

1. First and the recommended option is to use the *Bridged* type of network. This option exposes the guest machine to the same network that host network is connected to. The guest obtains network configuration information from a router or DHCP server on the network. The guest will appear as just another machine in the network. A major advantage of this of setup is that the host and all other machines in the network will be able to access the services provided by guest without requiring any further setup. The only drawback of this approach is that if the host is not connected to any network, the guest's network will remain unconfigured making it inaccessible even from the host.

2. Second method is *Host only* type of networking. With a guest's network interface configured in this manner, it will only be accessible from the host machine. The guest will not able access any other machine but the host, so you do not have internet access on the guest. All services on the guest are available to the host machine without any configuration such as port forwarding.

3. The third option is to use the *NAT* type of network. This the networking type that VirtualBox assigns to a freshly created virtual machine. This option works even when host is not connected to any network. The guest is automatically configured and is able to access the internet and local networks that host is able to connect to. However, the services provided by the guest require port forwarding configuration setup to be available outside.

   To configure this go to VM settings -> [Network] -> [Adapter] -> [Port Forwarding]. Map a port such as 2222 from host to guest port 22 and you will be able to ssh into FreedomBox from host machine as follows:

   ```
   ssh -p 2222 fbx@localhost
   ```

   Map 4443 on host to 443 on the guest. This make FreedomBox HTTPS service available on host using the URL https://localhost:4443/ You will need to add a mapping for each such services from host to guest.

4. The final option is to create two network interfaces, one *host only* and one *NAT* type. This way you can access the guest without any additional configuration, and you have internet access on the guest. The guest will be invisible to any other machines on the network.

Summary of various network types:

-	Guest accessible from other machines	Guest accessible from host	Works without port forwarding	Works without host connected to network	Guest has internet access
Bridged	✓	✓	✓	✗	✓
Host only	✗	✓	✓	✓	✗
NAT	✓	✓	✗	✓	✓
NAT and Host	✗	✓	✓	✓	✓

7.14.6.2 Finding out the IP address of the virtual machine

This depends on the network configuration you chose. With a *bridged adapter*, your virtual machine gets its IP address from the DHCP server of your network, most likely of your Router. You can try the first couple of IP addresses or check your router web interface for a list of connected devices.

If you chose *host-only adapter*, the IP address is assigned by the DHCP server of your VirtualBox network. In the VirtualBox Manager, go to File -> Preferences -> Network -> Host-only Networks. You can see and edit the DHCP address range there, typically you get assigned addresses close to the *Lower Address Bound*.

Another possibility of finding the IP address is to login via the Virtualbox Manager (or similar software). The FreedomBox images do not have any default user accounts, so you need to set an initial user and password using the passwd-in-image script.

See also QuickStart for instructions on how to scan your network to discover the IP of the VM.

7.14.6.3 Networking Problems with macchanger

The package `macchanger` can cause network problems with VirtualBox. If you have a valid IP address on your guest's host network adapter (like 192.168.56.101) but are not able to ping or access the host (like 192.168.56.1), try uninstalling `macchanger`:

```
$ dpkg --ignore-depends=freedombox-setup --remove macchanger
```

You might have to manually remove the script `/etc/network/if-prep-up/macchanger`. If Debian complains about unmet dependencies when you use a package manager (apt-get, aptitude, dpkg), try to remove 'macchanger' from the dependencies of 'freedombox-setup' in the file `/var/lib/dpkg/status`.

7.14.6.4 Mounting Images Locally

If you want to mount images locally, use the following to copy built images off the VirtualBox:

```
$ mkdir /tmp/vbox-img1 /tmp/vbox-root1
$ vdfuse -f freedombox-unstable_2013.0519_virtualbox-i386-hdd.vdi /tmp/vbox-img1/
$ sudo mount -o loop /tmp/vbox-img1/Partition1 /tmp/vbox-root1
$ cp /tmp/vbox-root1/home/fbx/freedom-maker/build/freedom*vdi ~/
$ sudo umount /tmp/vbox-root1
# $ sudo umount /tmp/vbox-img1 # corruption here.
```

7.14.6.5 Fixing the time after suspend and resume

The virtual machine loses the correct time/date after suspending and resuming. One way to fix this is to create a cron-job that restarts the time service `ntp`. You can add a crontab entry as root to restart ntp every 15 minutes by typing '`crontab -e`' and adding this line:

```
*/15 * *   *   *     /etc/init.d/ntp restart
```

Do not restart this service too often as this increases the load of publicly and freely available NTP servers.

7.14.6.6 UUID collision in VB

Whenever this happens VirtualBox shows following error message: *Cannot register the hard disk A with UUID ... because a hard disk B with UUID ... already exists in the media registry*

Creating several VMs from the same image causes collisions due to ID's (hostname, IP, UUID, etc) that are expected to be universally unique. Most can be handeled operating the running VM. But VirtualBox complains before that (at the very creation of the VM) about the hard disk's UUID. This is usual stuff when you develop/test e.g. FreedomBox.

You can change a clone's UUID in the terminal as follows:

```
$ VBoxManage internalcommands sethduuid path/to/the/hd/vdi/file
```

7.15 Debian

FreedomBox is a pure blend of Debian. This means that all the work on FreedomBox is available in Debian as packages. It also means that any machine running Debian can be turned into a FreedomBox.

This page describes the process of installing FreedomBox on a Debian system. Currently, FreedomBox works in Debian Stable (Buster), Testing (Bullseye), and Unstable (Sid).

Caution
Use a fresh Debian installation
Installing FreedomBox changes your Debian system in many important ways. This includes installing a firewall and regenerating server certificates. It is hence recommended that you install FreedomBox on a fresh Debian installation instead of an existing setup.

Caution
Console/GUI logins for non-admin users will be disabled
After FreedomBox is fully setup, your system will no longer allow users not belonging to the *admin* group to log in to the system via console, secure shell (SSH) or graphical login. This behaviour can be disabled from the Security page. Use the administrator account created during FreedomBox first boot for console logins and add further user accounts to *admin* group, if necessary.

7.15.1 Installing on Debian 10.0 (Buster) or newer

Check the Troubleshooting section below, for any tips or workarounds that might help during the install.

1. Install Debian 10.0 (Buster), or Unstable (Sid) on your hardware.
2. Update your package list.

   ```
   $ sudo apt-get update
   ```

3. Install `freedombox` package.

   ```
   $ sudo DEBIAN_FRONTEND=noninteractive apt-get install freedombox
   ```

   - The "DEBIAN_FRONTEND=noninteractive" will avoid several configuration prompts that would otherwise appear during the install.

4. During the installation, you will be provided a secret key that needs to be entered during the initial configuration process. Note this down. The secret can also be read at a later time from the file `/var/lib/plinth/firstboot-wizard-secret`.
5. You can start using FreedomBox. During initial wizard, you will need to enter the secret noted above.

7.15.2 Installing on Debian 9 (Stretch)

Check the Troubleshooting section below, for any tips or workarounds that might help during the install.

1. Install Debian 9 (Stretch) on your hardware.
2. Update your package list.

```
$ sudo apt-get update
```

3. Install `freedombox-setup` package.

```
$ sudo DEBIAN_FRONTEND=noninteractive apt-get install freedombox-setup
```

   - The "DEBIAN_FRONTEND=noninteractive" will avoid several configuration prompts that would otherwise appear during the install.

4. Run FreedomBox setup program. This installs further packages and sets up basic configuration.

```
$ sudo /usr/lib/freedombox/setup | tee freedombox-setup.log
```

   You may have to clear your existing network configuration. See Troubleshooting note #2 below.

5. Reboot the system. This is necessary to trigger the first-run script.

```
$ sudo reboot
```

6. After the system boots up, wait for it to reboot again. The first-run scripts sets up a few things and initiates a reboot.
7. After the second reboot you can start using FreedomBox.

7.15.3 Tips and Troubleshooting

1. There is a bug in policykit-1 package that causes errors and hangs during installation of freedombox-setup package. This bug is only applicable to Debian 9 (Stretch) and older. A workaround is to first install policykit-1 package and then reboot. After that, follow the above setup procedure.

```
$ sudo apt-get update
$ sudo apt-get install policykit-1
$ sudo reboot
```

2. FreedomBox uses NetworkManager to manage network configuration. If you have configured your network interfaces using Debian installer or by editing `/etc/network/interfaces`, FreedomBox will not manage those interfaces. (See bug #797614.) To let FreedomBox/NetworkManager manage your network interfaces, edit the `/etc/network/interfaces` manually and ensure that it contains only the following:

```
auto lo
iface lo inet loopback
```

   If you have already completed the setup process without doing this step, you will need to clear out the `/etc/network/interfa` file keeping only the above lines. Then perform a reboot. On Debian 9 (Stretch), after this network connections configured by the `setup` step above will configure your network. Network interfaces will then be in the `internal` or `external` firewall zone. This is essential for the FreedomBox's web interface to be reachable from other machines in the network. You can tweak network manager connections with the `nmtui` command if you wish.

3. FreedomBox will use an automatically configured IP address by default. You can assign a static IP address if necessary. Network configuration changes can be done using FreedomBox web interface or by using the `nmtui` or `nmcli` commands. `nmcli` can be used as follows:

```
nmcli con mod "Ethernet connection 1"  \
ipv4.addresses A.A.A.A/X  \
ipv4.gateway G.G.G.G  \
ipv4.dns N.N.N.N  \
ipv4.dns-search somedomain.com  \
ipv4.method "manual"  \
ipv4.ignore-auto-dns yes  \
ipv6.method ignore
```

4. ..with the block capitals and somedomain.com replaced with your actual address, mask description, gateway and dns server details.

7.16 DreamPlug

Caution
Deprecated Hardware
This hardware was supported earlier but is no longer supported. If you downloaded an earlier image and are running FreedomBox on this hardware, you will keep getting software updates. You can stay secure and up-to-date. However, no new images will be provided for this hardware. It is recommended that you migrate to newer, supported hardware using backup and restore.

DreamPlug is the hardware for which FreedomBox has been originally targeted. FreedomBox images are built and tested for it. For using this device as FreedomBox, a USB WiFi device that does not require non-free firmware is recommended.

You can find more support and discussion for DreamPlug on the official forum.

Recommendation: If you are using a board that uses SD cards, when you flash the FreedomBox image onto your SD card, we recommend that you use an SD card with at least 8GB of storage space.

7.16.1 Download

FreedomBox SD card images for this hardware are available. Follow the instructions on the download page to create a FreedomBox SD card and boot into FreedomBox. See also instructions for using an internal micro-SD with DreamPlug.

An alternative to downloading these images is to install Debian on DreamPlug and then install FreedomBox on it.

7.16.2 Networking

The network port towards the middle of the box, is configured by FreedomBox to be an upstream Internet link. The remaining port is configured for a local computer to connect to.

7.16.3 Firmware

Note that the factory firmware configurations may vary between revisions of the hardware, and render some images incompatible. See the DreamPlug firmware page for information on what images are compatible and how to update your DreamPlug firmware.

7.16.4 Build Image

FreedomBox images for this hardware can be built using Freedom Maker.

7.16.5 Testing

Instructions on how to test this hardware are available.

7.16.6 Availability

- Price: 159 USD
- DreamPlug manufacturer
- Reseller Spinifex in Australia

7.16.7 Hardware

- Open Hardware: No
- CPU: Marvell Kirkwood 88F6281 @ 1.2GHz
- RAM: 512MB 16bit DDR2-800 MHz
- Storage: 4 GB on board micro-SD
- Architecture: armel
- Ethernet: 2x 10/100/1000, RJ45
- WiFi: SD8787, 802.11 b/g/n
- SATA: eSATA 2.0 port

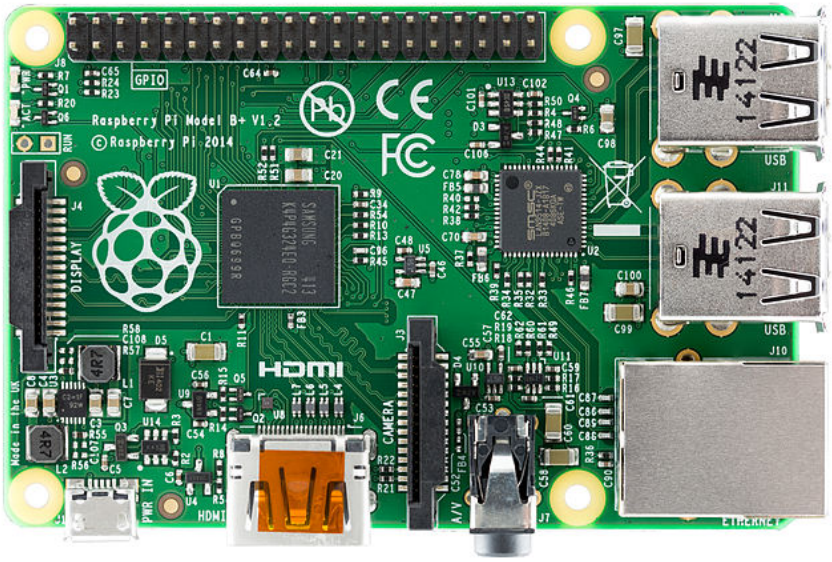
7.16.8 Non-Free Status

- Non-free blobs required: built-in WiFi
- WiFi: no free WiFi drivers + firmware available
- Works with stock Debian kernel: yes

7.16.9 Known Issues

- WiFi does not work with free software. A separate USB WiFi device is recommended.

7.17 Raspberry Pi Model B+



Caution
Deprecated Hardware
This hardware was supported earlier but is no longer supported. If you downloaded an earlier image and are running FreedomBox on this hardware, you will keep getting software updates. You can stay secure and up-to-date. However, no new images will be provided for this hardware. It is recommended that you migrate to newer, supported hardware using backup and restore.

Raspberry Pi (Model B+) is a popular single board computer developed with the intention of promoting teaching of basic computer science in schools. FreedomBox images are built and tested for it. For using this board as FreedomBox, a USB WiFi device that does not require non-free firmware is recommended.

Note: The Debian architecture used for this device is `armel`. This means floating point computations are done in software and most operations are slower than what Raspberry Pi is capable of.

Recommendation: When you flash the FreedomBox image onto your SD card, we recommend that you use an SD card with at least 8GB of storage space.

7.17.1 Download

FreedomBox SD card images for this hardware are available. Follow the instructions on the download page to create a FreedomBox SD card and boot into FreedomBox.

7.17.2 Build Image

FreedomBox images for this hardware can be built using Freedom Maker.

7.17.3 Availability

- Price: 35 USD
- List of official distributors

7.17.4 Hardware

- Open Hardware: No
- CPU: ARM1176JZF-S (ARMv6k) 700 MHz
- RAM: 512 MB
- Storage: MicroSD card slot
- Architecture: armel
- Ethernet: 10/100, RJ45
- WiFi: None, use a USB WiFi device
- SATA: None

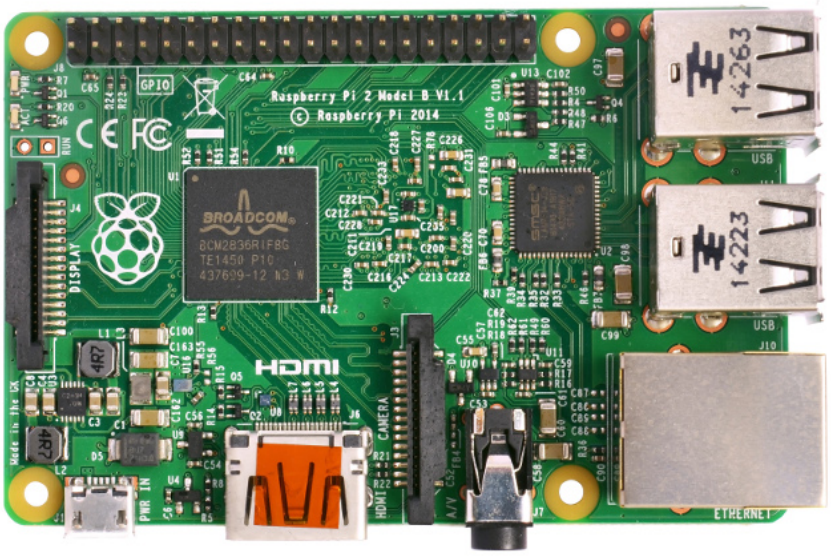
7.17.5 Non-Free Status

- Non-free blobs required: boot firmware
- WiFi: Not available
- Works with stock Debian kernel: No

7.17.6 Known Issues

- The Debian architecture used for this device is `armel`. This means floating point computations are done in software and generally most operations are slower than what Raspberry Pi is capable of.

7.18 Raspberry Pi 2 Model B



Raspberry Pi 2 (Model B) is a popular single board computer developed with the intention of promoting teaching of basic computer science in schools. It is a successor to Raspberry Pi Model B+ with much faster processor and more RAM. FreedomBox images are built and tested for it. For using this board as FreedomBox, a USB WiFi device that does not require non-free firmware is recommended.

Please do not expect any output on a monitor connected via HDMI to this device as it does not display anything beyond the message 'Starting kernel...'. See the Quick Start page to access and control your FreedomBox from network.

Recommendation: If you are using a board that uses SD cards, when you flash the FreedomBox image onto your SD card, we recommend that you use an SD card with at least 8GB of storage space.

7.18.1 Download

FreedomBox SD card images for this hardware are available. Follow the instructions on the download page to create a FreedomBox SD card and boot into FreedomBox.

7.18.2 Build Image

FreedomBox images for this hardware can be built using Freedom Maker.

7.18.3 Availability

- Price: 35 USD
- List of official distributors

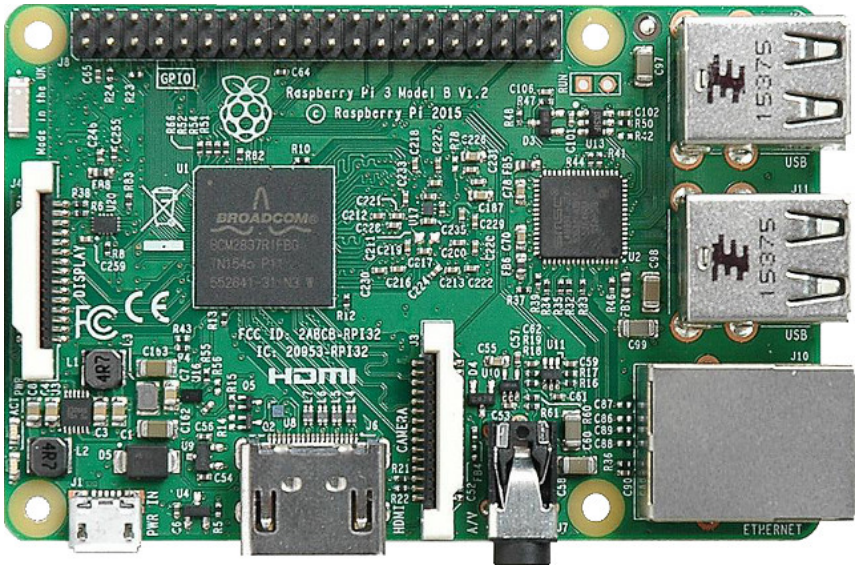
7.18.4 Hardware

- Open Hardware: No
- CPU: 900 MHz quad-core ARM Cortex-A7
- RAM: 1 GB
- Storage: MicroSD card slot
- Architecture: armhf
- Ethernet: 10/100, RJ45
- WiFi: None, use a USB WiFi device
- SATA: None

7.18.5 Non-Free Status

- Non-free blobs required: boot firmware
- WiFi: Not available
- Works with stock Debian kernel: Yes

7.19 Raspberry Pi 3 Model B



Raspberry Pi 3 Model B is a popular single board computer developed with the intention of promoting teaching of basic computer science in schools. It is a successor to Raspberry Pi 2 Model B with a 64-bit processor and on-board Wi-Fi. A FreedomBox "testing" image is available for Raspberry Pi 3 Model B. For using this board as FreedomBox, a USB WiFi device, that does not require non-free firmware, is recommended instead of the on-board Wi-Fi.

Please do not expect any output on a monitor connected via HDMI to this device as it does not display anything beyond the message 'Starting kernel...'. See the Quick Start page to access and control your FreedomBox from network.

Recommendation: If you are using a board that uses SD cards, when you flash the FreedomBox image onto your SD card, we recommend that you use an SD card with at least 8GB of storage space.

7.19.1 Download

FreedomBox SD card images for this hardware are available. Download the "testing" image for Raspberry Pi 3 Model B. Follow the instructions on the download page to create a FreedomBox SD card and boot into FreedomBox.

7.19.2 Build Image

FreedomBox images for this hardware can be built using Freedom Maker. Use the target 'raspberry3' with distribution 'testing' to build the image for this board.

7.19.3 Availability

- Price: 35 USD
- List of official distributors

7.19.4 Hardware

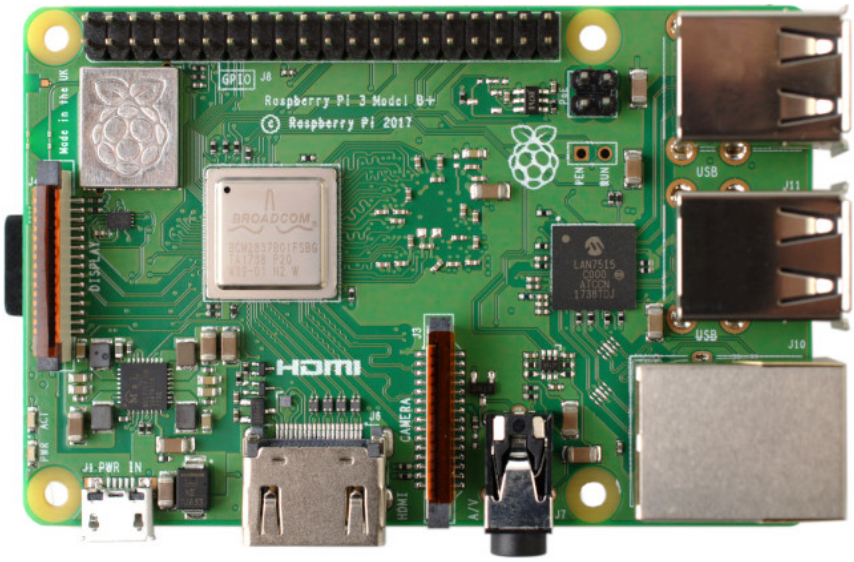
- Open Hardware: No
- CPU: 1.2GHz 64-bit quad-core ARMv8 CPU
- RAM: 1 GB
- Storage: MicroSD card slot

- Architecture: armhf
- Ethernet: 10/100, RJ45
- WiFi: 802.11n but requires non-free firmware, instead use a USB WiFi device
- SATA: None

7.19.5 Non-Free Status

- Non-free blobs required: boot firmware
- WiFi: Requires non-free firmware
- Works with stock Debian kernel: Yes

7.20 Raspberry Pi 3 Model B+



Raspberry Pi 3 Model B+ is a popular single board computer developed with the intention of promoting teaching of basic computer science in schools. It is a successor to Raspberry Pi 3 Model B with better Ethernet and a 5Ghz Wi-Fi. A FreedomBox "testing" image is available for Raspberry Pi 3 Model B+. For using this board as FreedomBox, a USB WiFi device, that does not require non-free firmware, is recommended instead of the on-board Wi-Fi.

Please do not expect any output on a monitor connected via HDMI to this device as it does not display anything beyond the message 'Starting kernel...'. See the Quick Start page to access and control your FreedomBox from network.

Recommendation: If you are using a board that uses SD cards, when you flash the FreedomBox image onto your SD card, we recommend that you use an SD card with at least 8GB of storage space.

7.20.1 Download

FreedomBox SD card images for this hardware are available. Download the "testing" image for Raspberry Pi 3 Model B+. Follow the instructions on the download page to create a FreedomBox SD card and boot into FreedomBox.

7.20.2 Build Image

FreedomBox images for this hardware can be built using Freedom Maker. Use the target 'raspberry3-b-plus' with distribution 'testing' to build the image for this board.

7.20.3 Availability

- Price: 35 USD
- List of official distributors

7.20.4 Hardware

- Open Hardware: No
- CPU: 1.4GHz 64-bit quad-core ARMv8 CPU
- RAM: 1 GB
- Storage: MicroSD card slot
- Architecture: armhf
- Ethernet: 10/100/1000, RJ45
- WiFi: 802.11ac but requires non-free firmware, instead use a USB WiFi device
- SATA: None

7.20.5 Non-Free Status

- Non-free blobs required: boot firmware
- WiFi: Requires non-free firmware
- Works with stock Debian kernel: Yes

7.21 USB Wi-Fi

FreedomBox works on many single board computers. However, many of these boards do not have built-in Wi-Fi capabilities. Even when Wi-Fi capability is available, non-free proprietary firmware is required to make them work.

A solution to the problem is to plug-in a USB Wi-Fi device into one of the available USB ports. There are many such devices available which do not require non-free firmware to work. The following is a list of such devices that work with FreedomBox devices. Some devices based on these chips have tested to work well with FreedomBox including functions such as access point mode.

- Devices with Atheros AR7010 chip
- Devices with Atheros AR9271 chip

7.21.1 Firmware Installation

The free firmware for these devices is not packaged in Debian yet. You can manually download and install the firmware as follows:

```
sudo su [enter password]
cd /lib/firmware
wget https://www.thinkpenguin.com/files/ath9k-htc/version-1.4-beta/htc_9271.fw
wget https://www.thinkpenguin.com/files/ath9k_firmware_free-version/htc_7010.fw
```

7.21.2 Resources

- Debian Wiki on WiFi drivers
- Wikipedia: Comparison of open-source Linux wireless network drivers
- WikiDevi: database of computer hardware

7.22 Release Notes

The following are the release notes for each FreedomBox version.

7.22.1 FreedomBox 20.7 (2020-04-20)

- matrixsynapse: Fix initial installation and upgrade from backports
- gitweb: Improve error handling when creating repository
- locale: Update translations for French, Serbian, and Telugu

7.22.2 FreedomBox 20.6.1 (2020-04-11)

- users: Restore line of help text that was accidentally dropped
- debian: Add firmware-ath9k-htc to Recommends
- gitweb: Use proper ellipsis char when showing clone progress
- locale: Update translations for Norwegian Bokmål, German, French, Portuguese, Italian, Russian, and Serbian

7.22.3 FreedomBox 20.6 (2020-04-06)

- app: Ensure toggle buttons work independently of configuration form
- networks, monkeysphere: Make styling more specific to avoid interference
- syncthing: Update description to mention 'syncthing' group
- radicale: Support upgrade up to any 2.x version
- packages: Hold freedombox package during package installs
- users: Add component for managing users and groups
- app: Fix grammar in developer documentation string
- ikiwiki: Disable public edits of blog pages
- ikiwiki: Add moderation of blog comments
- firewalld: Support upgrade up to any 0.8.x version
- infinoted: Fix permissions of sync directory
- locale: Added Serbian translation
- locale: Update translations for Russian, French, German, Czech, Italian, Hindi, Telugu, and Spanish

7.22.4 FreedomBox 20.5.1 (2020-03-26)

- networks: Update label wording in topology form
- jsxc: Fix issue with serving static files
- debian: Separate binary packages for each language manual
- locale: Update translations for Norwegian Bokmål and German

7.22.5 FreedomBox 20.5 (2020-03-23)

- app: Fix description block in app header
- pagekite: Don't signal new domain on init if app is disabled
- pagekite: Don't attempt to notify about domain if app is disabled
- pagekite: Remove app enabled checking from getting configuration
- pagekite: On enable/disable, add/remove domain from names module
- pagekite: Fix an error message in custom services form
- matrixsynapse: Handle release of matrix-synapse 1.11
- setup: Fix regression to force-upgrade caused by Info changes
- pagekite: Don't allow non-unique custom services
- index: Reintroduce clients button in front page
- upgrades: Don't ship apt backport preferences file
- upgrades: Use internal scheduler instead of systemd timer
- shadowsocks: Change default configuration
- shadowsocks: Fix incorrect setting of state directory
- shadowsocks: When editing configuration, don't re-enable
- mediawiki: Don't allow anonymous edits
- names: Fix Local Network Domain is not shown
- shadowshocks: Fix setting configuration on Buster
- locale: Update translations for Swedish, Spanish, and French

7.22.6 FreedomBox 20.4 (2020-03-09)

- apache: Handle transition to php 7.4
- app: Fix showing app name in port forwarding information
- apps: Do not show status block if service is running
- i2p: New style app page layout
- locale: Update translations for French, Telugu, Spanish, and Swedish
- networks: Add first boot step for network topology wizard
- networks: Add form for network topology

- networks: Don't show router wizard if not behind a router
- networks, firewall: Support newer version of policykit
- networks: Fixes for networks wizards access and user experience
- networks: If topology wizard is skipped, skip router wizard too
- networks: Show router wizard before Internet connection type wizard
- plinth: Increase sqlite busy timeout from default 5s to 30s
- quassel: Fix unable to disable application without choosing a domain name
- shadowsocks: Move user settings to state directory
- storage: Directory selection form improvements
- transmission: Allow to submit download directory if it is creatable
- upgrades: Clean apt cache every week
- views: Improve template security

7.22.7 FreedomBox 20.3 (2020-02-24)

- apps: Update style for toggle button
- apps: Drop border shadow for app icon in mobile view
- apps: Show short description as secondary title
- apps: Remove css filters and glow from app icons
- cards: Remove the transition delay on hover effect
- system: Implement new style for cards
- framework: Generate secret key (existing sessions will get logged out)
- framework: Cleanup expired sessions every week
- networks: Add setting for internet connection type
- networks: Ask about internet connection type during setup
- shadowsocks: Fix shadowsocks not able to start
- jsxc: Bypass issue with stronghold to get the app working again
- monkeysphere: Fix regression with reading Apache configuration
- help: Fix attribute on download manual button
- firewall: Improve speed of some operations using DBus API
- css: Add missing license identifier on some CSS files
- deluge: Use safer method for editing configuration
- deluge: More reliable initial configuration setup
- samba: Add link to manual page
- searx: Update search engines for 0.16.0
- openvpn: Fix spelling for Tunnelblick
- bind: Show served domains
- Update translations for German, Swedish, Italian, Spanish, Norwegian Bokmål, Hungarian, Polish, and French

7.22.8 FreedomBox 20.2 (2020-02-10)

- networks: Support virtual Ethernet (veth) devices
- diagnostics: Show firewall service status
- storage: Show disks if FreedomBox is running in an unprivileged container
- service: Stop service not before but after disabling it
- users: Use more precise username validation
- sso, users: Turn off autocapitalization on the username field
- help: Fix anchor hidden under navbar
- searx: Fix installation issue for 0.16.0
- firewall: Show Run Diagnostics button in app
- glib: Introduce method to schedule an operation at regular intervals
- notification: Show a drop down from main navbar for notifications
- storage: Show low disk space warning using notifications API
- upgrades: Show notification when FreedomBox is updated
- security: Add Sandbox Coverage to report page
- matrixsynapse: Enable systemd sandboxing
- locale: Update translations for Telugu, French, Norwegian Bokmål, German, Spanish, and Swedish

7.22.9 FreedomBox 20.1 (2020-01-27)

- deluge: Allow to set a download directory
- deluge: Fix installation failure on slow machine
- storage: Make external disk mounts accessible to other users
- gitweb: Add link to the manual page
- style: Fix incorrect margins for containers in mobile view
- style: Fix responsiveness for app header
- network: Fix activating connections that don't have real devices
- wireguard: Add WireGuard VPN app
- networks: Add router configuration page
- networks: Add first boot step for router config helper
- bind: Enable sandboxing for bind service
- locale: Updated translations for Dutch, Norwegian Bokmål, German, Spanish, Swedish, French, and Greek

7.22.10 FreedomBox 20.0 (2020-01-13)

- samba: Improve speed of actions
- deluge: Manage deluged service and connect automatically from web interface
- openvpn: Enable support for communication among all clients
- storage: Ignore errors resizing partition during initial setup
- storage: Make partition resizing work with parted 3.3
- debian: Add powermgmt-base as recommended package
- openvpn: Enable IPv6 for server and client outside the tunnel
- networks: Fix crashing when accessing network manager D-Bus API
- mediawiki: Use a mobile-friendly skin by default
- mediawiki: Allow admin to set default skin
- matrixsynapse: Allow upgrade to 1.8.*
- security: Add explanation of sandboxing
- Update translations for Greek, German, Swedish, Hungarian, Norwegian Bokmål, and French

7.22.11 FreedomBox 19.24 (2019-12-30)

- app: Fix JavaScript doesn't run on first visit
- samba: Add private shares
- firewall: Support upgrading firewalld to 0.8
- deluge: Add systemd sandboxing features
- infinoted: Add systemd sandboxing features
- storage: Add systemd sandboxing features to udiskie service
- upgrades: Add systemd sandboxing features to repository setup service
- security: List whether each app is sandboxed
- mediawiki: Avoid delay in update script
- diagnostics: Use new component based API for all diagnostic tests
- minidlna: Fix showing clients information
- mediawiki: Fix problem with session cache failing logins
- locale: Update translations for French, German, Swedish, Greek, Hungarian, Norwegian Bokmål, and Dutch

7.22.12 FreedomBox 19.23 (2019-12-16)

- minidlna: New app for MiniDLNA (Simple Media Server)
- apps: Show app icons in app pages
- apps: Implement responsive layout for app pages
- samba: Recursively set open share directory permissions
- transmission: Add directory selection form
- mumble: Add option to set SuperUser password
- cockpit: Extend apps description with access info
- cockpit: Add list of valid urls to access the app
- Update translations for French, German, Spanish, Portuguese, and Swedish

7.22.13 FreedomBox 19.22 (2019-12-02)

- samba: Add new app for Samba file sharing
- pagekite: Remove tabs in the configuration page
- openvpn: Fix text with manual link
- pagekite: Show existing services only if there are any
- pagekite: Move Custom Services under Configuration
- pagekite: Use the new app toggle button
- openvpn: Add client apps
- backups: Fix title not appearing
- diagnostics: Don't run on disabled modules
- apps: Remove link to webapps in app descriptions
- interface: Fix error with app toggle input
- templates: Add toolbar for apps
- toolbar: Move diagnostics button into dropdown menu
- ssh: Fix Avahi SFTP service file
- diagnostics: Fix IPv6 failures
- matrix-synapse: Fix installation of 1.5 from buster-backports
- app: Fix javascript constant redeclaration error
- ikiwiki: Move the create button to manage section
- gitweb: Move create button into manage section
- networks: Move actions button into connection section
- users: Move create button into users section
- locale: Update translations for French, German, and Swedish

7.22.14 FreedomBox 19.21 (2019-11-18)

- gitweb: Allow to import from a remote repository
- interface: Disable turbolinks on links that don't point to /plinth/...
- backups: Show proper error when SSH server is not reachable
- tor: Rename "Hidden Service" to "Onion Service"
- ejabberd: Handle case where domain name is not set
- tahoe: Mark Tahoe-LAFS as an advanced app
- searx: Set safe_search to Moderate by default
- backups: Make verify ssh host page string translatable
- backups: Simplify SSH fingerprint verification command
- doc: Fix unavailability of manual images
- tor: Fix port diagnostics by correcting port data type
- tor: Expect obfs service to be also available on IPv6
- tor: Listen on IPv6 for OrPort
- clients: implement launch button feature
- apps: Implement toggle button in apps pages
- Update translations for German, Hungarian, Swedish, Norwegian Bokmål, French, Polish

7.22.15 FreedomBox 19.20 (2019-11-04)

- doc: Add Spanish manual
- ssh: Add option to disable password authentication
- sharing: Fix wrong links on Apache2 directory index page
- gitweb: Set correct access rights after enabling application
- gitweb: Fix links leading to blank page
- gitweb: Set proper access after restoration of a backup
- snapshot: Sort snapshot list from newest to oldest
- infinoted: Add missing manual page link
- backups: Fix typo
- Update translations for German, Spanish, Swedish, Czech, French, Norwegian Bokmål, Hungarian

7.22.16 FreedomBox 19.19 (2019-10-21)

- gitweb: New app for simple git hosting
- ikiwiki: Allow full Unicode text in wiki/blog title names
- users: reload Apache2 to flush LDAP cache after user operations
- ssh: Show server fingerprints in SSH page
- frontpage: Show public shortcuts to all users regardless of group
- ikiwiki: Remove extra create button when no wiki/blog is present
- quassel: Add Let's Encrypt component for certificates
- Update translations for Czech, French, Bulgarian, Dutch, German, and Norwegian Bokmål

7.22.17 FreedomBox 19.18 (2019-10-07)

- diagnostics: Ensure that exceptions are reported as failures
- users: Rearrange UI to match with other apps
- upgrades, ikiwiki, networks, backups: Replace page tabs with buttons
- dynamicdns, i2p, pagekite, snapshot: Cleanup page templates
- deluge: Support deluge 2 by starting it properly
- minetest: Remove mod-torches no longer available in testing/unstable
- security: Add past vulnerabilities count, move report to new page
- Update translations for Spanish, Norwegian Bokmål, German

7.22.18 FreedomBox 19.17 (2019-09-23)

- firstboot: Add new help menu to firstboot navbar
- firstboot: Hide left menu during first boot as intended
- Update translations for Chinese (Simplified) and Czech
- Fix tests for letsencrypt and tor

7.22.19 FreedomBox 19.16 (2019-09-09)

- backups: Allow adding backup repositories on multiple disks
- help: Add buttons for contribute, support, and feedback
- action_utils: Workaround problem with setting debconf answers
- views: Fix failure in redirecting from language selection page
- manual: Move PDF download link to HTML manual page
- help: Convert help icon in the navbar to dropdown
- ejabberd: Fix listen port configuration for ejabberd 19.x
- cockpit, ejabberd: Prevent restart on freedombox startup
- ejabberd: Perform host/domain name operations only when installed
- logging: Improve formatting and reduce noise
- translations: Update Hungarian, German, Italian, French, and Norwegian Bokmål

7.22.20 FreedomBox 19.15 (2019-08-26)

- security: Hide vulnerability table by default
- names: Perform better layout of domain names table on small screens
- cockpit: Apply domain name changes immediately
- ejabberd: Prevent processing empty domain name
- config: Send hostname change signal only after fully processing it
- letsencrypt: Don't try to obtain certificates for .local domains
- avahi: Expose .local domain as a proper domain
- cockpit: Make essential and install by default
- tt-rss: Force upgrade to 18.12-1.1 and beyond
- updates: Allow matrix-synapse 1.3 to be installed for buster users
- javascript: Don't resubmit when refreshing the page
- storage: Fix regression with restoring backups with storage
- matrix-synapse: Use recommended reverse proxy configuration
- Update translations for German, Hungarian, and Norwegian Bokmål

7.22.21 FreedomBox 19.14 (2019-08-12)

- storage: Handle all device paths during eject
- storage: Fix incorrect internationalization when throwing an error
- upgrades: Use collapsible-button style for logs
- firewall: Allow automatic upgrade to 0.7.x
- upgrades: Handle release info change
- frontpage: Fix regression with loading custom shortcuts
- names: Add dynamic domain name
- names: Add button to configure each type of name
- names: Update page layout for clearer presentation
- names: Introduce new API for domain name handling
- api: Fix regression with listing only enabled apps in mobile app
- Update translations for Czech, Hungarian, French, Chinese (Simplified), Turkish, Polish, and Norwegian Bokmål

7.22.22 FreedomBox 19.13 (2019-07-29)

- backups: Make UI more consistent with other apps
- backups: Make backup location tables collapsible
- Updated translations for Chinese (Simplified), German, and Norwegian Bokmål
- help: Show security notice when backports are in use
- security: Show vulnerability counts

7.22.23 FreedomBox 19.12 (2019-07-22)

- sharing: Allow directories to be publicly shared
- backups: Add option to select/deselect all apps for backup or restore
- dbus: Allow plinth user to own FreedomBox DBus service
- letsencrypt: Simplify renewal hooks implementation
- cockpit: Don't handle domains if app is not installed
- dynamicdns: Send domain added signal properly during init
- ejabberd: Backup and restore TLS certificates
- Started new Galician translation on Weblate
- Updated translations for Czech, Norwegian Bokmål, Hungarian, Spanish, Telugu, Chinese (Simplified), German, Turkish, and Russian

7.22.24 FreedomBox 19.2.2 (2019-07-17)

This release does not contain any functional changes, but fixes test failures when building the package.

7.22.25 FreedomBox 19.2.1 (2019-07-09)

This is a bugfix release for 19.2.

- dbus: Allow plinth user to own FreedomBox DBus service

7.22.26 FreedomBox 19.11 (2019-07-08)

- backups: Fixes to issues while adding SSH remotes:
  - Improve UX of adding ssh remote
  - Avoid creating duplicate SSH remotes
  - Fix issue with repository not being initialized
  - Verify SSH hostkey before mounting
  - Allow SSH directory paths with : in them
  - Require passphrase for encryption in add repository form
  - Don't send passphrase on the command line
  - Un-mount SSH repositories before deleting them
- matrixsynapse: Fix missing translation mark
- Started new Greek translation on Weblate
- Updated translations for Chinese (Simplified), Hungarian, Spanish, and Russian

7.22.27 FreedomBox 19.10 (2019-06-24)

- syncthing: Open firewall ports for listening and discovery
- radicale: Workaround issue with creating log directory
- Update translations for Turkish, German, Czech, Norwegian Bokmål, and Portuguese
- Introduce components for firewall, webserver, uwsgi, and daemons

7.22.28 FreedomBox 19.9 (2019-06-10)

- config: Add option to show advanced apps, which are hidden by default
- monkeysphere: Hide by default
- searx: Add option to allow public access to the application
- Introduce component architecture for apps, with components for menus and shortcuts
- Start new translation for Bulgarian
- Update translations for Turkish and Norwegian Bokmål

7.22.29 FreedomBox 19.8 (2019-05-27)

- Switch to using SVG icons for all apps.
- Updated translations for Czech, Norwegian Bokmål, Hungarian, German, Turkish, and Spanish.

7.22.30 FreedomBox 19.7 (2019-05-13)

- i2p: Include default favorites.
- Separate enabled and disabled apps.
- Display port forwarding info for apps.
- Added Slovenian translation.
- Updated translations for Dutch, German, Hungarian, Norwegian Bokmål, Polish, Portuguese, Telugu.

7.22.31 FreedomBox 19.6 (2019-04-29)

- i2p: Enable new application for I2P Anonymity Network.
- Updated translations for Czech, German, Norwegian Bokmål, and Turkish.
- letsencrypt: Provide link to configure domain if not configured.
- firewall: Show port numbers and types.

7.22.32 FreedomBox 19.5 (2019-04-15)

- storage: Use more reliable method to list disks and disk space usage.
- Updated translations for Russian and German.

7.22.33 FreedomBox 19.4 (2019-04-01)

- clients: Open web app in a new browser tab
- matrix-synapse: Change client diagnostics url
- minetest: Fix duplicate domain names being displayed in UI
- storage: Do not show an eject button on /boot partitions
- letsencrypt: Call letsencrypt manage_hooks with correct arguments

- dynamicdns: Install module by default
- storage: Don't check type of the disk for / and /boot
- storage: Don't log error when checking if partition is expandable
- Updated translations for Norwegian Bokmål, Czech, German, Hungarian, Spanish, German, and Russian.

7.22.34 FreedomBox 19.3 (2019-03-18)

- UI: Move tabs below descriptions.
- firewall: Style heading
- names: Add description
- pagekite: Change heading text
- ikiwiki: Consistent styling for delete warning page
- main: Show service version in logs
- setup: Organize data files into various apps
- Updated translations for Czech, Hungarian, Norwegian Bokmål, Spanish, German, French, Italian, and Turkish.

7.22.35 FreedomBox 19.2 (2019-03-02)

- config: Fix Ikiwiki entries not showing up as default apps
- config: Migrate default app configuration to new conf file
- config: Rename Default App to Webserver Home Page
- config: Add option to use Apache's default home page as home page
- config: Fix error when setting JSXC as the home page
- Disable Coquelicot for Buster release
- matrix-synapse: Fix LDAP login issue
- config: Revert changes in freedombox.conf to avoid conffile prompt
- openvpn: Migration from easy-rsa 2 to 3 for existing installations
- tor: Use fixed 9001 port for relaying
- package: Implement identifying packages that need conffile prompts
- setup: Trigger force upgrade for app that implement it
- bind: Handle conffile prompt during upgrade
- apache: Pre-enable necessary apache modules
- apache: Use cgid module instead of cgi
- openvpn: Make frontpage shortcut appear after an upgrade
- openvpn: Work around firewalld bug 919517
- firewalld: Implement upgrading from 0.4.x to 0.6.x
- ttrss: Implement upgrade from 17.4 to 18.12

- radicale: Add description of web interface
- ttrss: Add backup support
- security: Migrate access config to new file
- Updated translations for Czech, Hungarian, Norwegian Bokmål, Spanish, German, Telugu.

7.22.36 FreedomBox 19.1 (2019-02-14)

- radicale: Increment module version to trigger upgrade handling
- radicale: Remove obsolete diagnostics
- radicale: Fix server URLs in client info
- Updated translations for Czech, Norwegian Bokmål, and Spanish.
- setup: Add option to handle configuration prompts during install
- radicale: Simplify upgrading to newer packages
- matrixsynapse: Use Let's Encrypt certificates

7.22.37 FreedomBox 19.0 (2019-02-09)

- mldonkey: Add some more clients to the module page
- mldonkey: Add to the description the three available front-ends
- monkeysphere: Fix handling of multiple domains and keys
- monkeysphere: Fix regression with reading new apache domain config
- apache: Switch to mod_ssl from mod_gnutls
- mldonkey: Enable app
- upgrades: Fix priority for buster-backports version
- upgrades: Fix premature adding of buster-backports sources
- Updated translations for Czech, German, and Spanish
- Switched to a new version number scheme: YY.N
  - YY is the year of release.
  - N is the release number within that year.

7.22.38 Version 0.49.1 (2019-02-07)

- ui: Fix regression with configure button in home page.
- backups: Rename 'Abort' buttons to 'Cancel'.
- backups: Use icon for add repository button.
- backups: Move subsubmenu below description.
- backups: Add title and description to other pages.
- backups: Add link to manual page.

- backups: Fix styling for upload size warning.
- backups: Increase timeout for SSH operations to 30 seconds.
- letsencrypt: UI: Fix checkbox disabling.
- datetime: Switch from chrony to systemd-timesyncd.
- Updated translations for Czech, Norwegian Bokmål, and Spanish.

7.22.39 Version 0.49.0 (2019-02-05)

- security: Update javascript for Content Security Policy.
- help: Use correct package to determine available version.
- repro: Disable app due to issues with Debian package.
- ui: Fix regression with card icon style in front page.
- js: Support full librejs compatibility.
- js: Remove javascript license link from footer.
- backups: Remove incorrectly set buffer size during download.
- backups: Fix incomplete download archives.
- backups: Improve performance of backup download.
- radicale: Handle migration from 1.x to 2.x.
- datetime: Switch from ntp to chrony.
- backports: Add buster-backports to apt sources list.
- Updated translations for Czech, Norwegian Bokmål, and Hungarian.

7.22.40 Version 0.48.0 (2019-01-28)

- Updated translations for Czech, Hungarian, German, and Norwegian Bokmål.
- UI improvements:
  - Fix top margin for content containers.
  - Fix setting width of card-list at various page sizes.
  - Show help nav item text when navbar is collapsed.
  - Hide restart/shutdown items when navbar is collapsed.
  - Compact pages on extra small screen sizes.
- Backups improvements:
  - Add backup/restore support for syncthing and openvpn.
  - Upgrade apps before restoring them
  - Fix showing not-installed apps in create backup page
  - Automatically install required apps before restore.
  - Add a loader to the restore button to indicate progress.
- Serve default favicon for apps that don't provide one.
- radicale: Fix issue with configuration changes not applying.

- storage: Fix false error message in log when visiting home page.
- infinoted: Handle timeout issue when stopping daemon during setup.
- matrix-synapse: Fix startup error caused by bind_address setting.
- radicale: Avoid changes to conffile for radicale 2.x.
- help: Fix showing status logs when an error occurs.
- fail2ban: Enable bans for apache auth failures.
- mldonkey: Initial work on new module for the eDonkey network.
  - Not available yet, due to bug in package.

7.22.41 Version 0.47.0 (2019-01-14)

- Show Gujarati in the list of languages.
- Replace glyphicons with forkawesome icons.
- Snapshots:
  - Change configuration to avoid filling up disk.
  - Handle "Config in use" error.
  - Update descriptions and configuration options.
- Firewall: Fix issue with transition from iptables.
- Security: Switch to Argon2 password hash.
- Cockpit: Add link to manual page and update description.
- Radicale: Add initial support for radicale 2.x.
- Setup:
  - Handle showing setup page after app completes installation.
  - Optimize installation in-progress checks and refresh time.

7.22.42 Version 0.46.0 (2018-12-31)

- Updated translations for Czech, German, Spanish, Ukrainian, and Norwegian Bokmål.
- Use systemd journal for logging.
- Rename plinth binary package to "freedombox", and merge freedombox-setup package into it.

7.22.43 Version 0.45.0 (2018-12-17)

- Storage: Merge list of removable media into existing table.
- Backups: Allow remote backups to SSH servers using sshfs.
- Backups: Removed asking for backup archive name.
- Automatically handle future versions of PHP.
- Updated translations for Hungarian, Czech, Spanish, Chinese (Simplified), Italian, Norwegian Bokmål, French, and German.

7.22.44 Version 0.44.0 (2018-12-03)

- UI: Add card style and gray noise background to apps pages.
- UI: Fix distortion of the client apps buttons.
- ejabberd: Handle BOSH port change from TCP 5280 to 5443.
- Minetest: Update mods list to available Debian packages.
- Firewall: Use nftables instead of iptables.
- Snapshots: Fix default snapshot listing.
- Snapshots: Show description above either tab.
- Snapshots: Allow snapshots to be selected for deletion.
- Translations: Updated Czech, Norwegian Bokmål, Spanish, German, and Portuguese.

7.22.45 Version 0.43.0 (2018-11-19)

- Backups improvements:
  - Allow backups to be downloaded directly, without export step.
  - Restore directly from uploaded backup.
  - Avoid error for apps with no data to backup.
  - Show free disk space on upload and restore page.
  - Do not limit maximum upload size.
- openvpn: Migrate to easy-rsa 3 and fix setup issues.
- Make single sign-on tickets valid for 12 hours.
- Use consistent terminology for updates.
- Updated translations for Czech and Portuguese.

7.22.46 Version 0.42.0 (2018-11-05)

- Fix wrong color in mobile menu
- snapshot: Fix broken snapshot management after snapper update
- Enable backup/restore for tor, upgrades, monkeysphere, letsencrypt, tahoe
- monkeysphere: Handle importing new OpenSSH format keys
- udiskie: unmount drive as superuser
- Updated translations for Telugu, Indonesian, and Italian

7.22.47 Version 0.41.0 (2018-10-22)

- Enable backup/restore for datetime, deluge, avahi, backups, bind, security, snapshot, ssh, firewall, diagnostics, names, power, and storage.
- snapshot: Fix issue with setting configuration.
- backups: Fix backup archives ownership issue.
- backups: Fix issue with showing exports from disks without labels.
- backups: Don't rely on disk labels during export/restore.
- backups: Fix downloading extracted archive files.
- Updated translations for Norwegian Bokmål, French, Russian, and Spanish.

7.22.48 Version 0.40.0 (2018-10-08)

- Backups
  - Enable backup/restore for mumble, privoxy, roundcube, searx, jsxc, coquelicot, transmission, quassel, shadowsocks, sharing, pagekite, and cockpit.
  - Allow backup archives to be downloaded/uploaded through browser.
  - mediawiki: Backup/restore settings as well as data.
- User Interface
  - Change card text style and position.
  - Change maximum cards per row.
  - Add tint effect on card icons under "Apps".
- mediawiki: Run update script for 1.31 upgrade.
- customization: Show custom shortcuts on frontpage.
- Updated translations for Norwegian Bokmål, Portuguese, Spanish, Czech, German, French, and Italian.

7.22.49 Version 0.39.0 (2018-09-24)

- Updated translations for Hungarian and Norwegian Bokmål.
- Merge Removable Media (udiskie) into Storage module.
- Add Backups module for backing up apps data.

7.22.50 Version 0.38.0 (2018-09-10)

- mediawiki: Enable SVG support for MediaWiki
- upgrades: Clean up old kernel packages during automatic upgrades
- Make the progress bar at the top of the page more visible.
- Updated translations for Norwegian Bokmål, Czech, Russian, German, Hungarian, and Spanish.

7.22.51 Version 0.37.0 (2018-08-27)

- Updated translations for Czech, Norwegian Bokmål, Russian, Spanish, Hungarian, and Dutch.
- install: Use Post/Response/Get pattern for reloads.

7.22.52 Version 0.36.0 (2018-08-13)

- Updated translations for Hindi, Spanish, Russian, Telugu, German, Hungarian, Czech, and French
- ejabberd: Remove deprecated settings from already existing config files
- mediawiki: Fix issue with re-installation
- mediawiki: Enable Instant Commons
- mediawiki: Fix images throwing 403s
- turbolinks: Reload page using JavaScript
- Add Lato woff2 fonts
- Disable launch button for web client when not installed

7.22.53 Version 0.35.0 (2018-07-30)

- configuration: Add an option to set a default app for FreedomBox. The root URL path (`https://domainname/`) will redirect to the selected app.
- ejabberd: Remove deprecated `iqdisc` setting. To apply this fix, disable and then re-enable the Message Archive Management setting.
- ejabberd: Replace logo with original version.
- mediawiki: Enable short URLs, which look like `https://domainname/mediawiki/ArticleName`.
- radicale: Clarify description for shared calendar/addressbook.
- storage: Handle mount points with spaces.
- udiskie: Add button to eject drives.
- udiskie: Also show read-only filesystems.
- udiskie: Remove internal networks warning.
- udiskie: Show special message when no storage device available.
- Add turbolinks library for smoother navigation.
- Removed extra text from icons for mediawiki, radicale, and tahoe-lafs.
- Updated translations for Russian, Spanish, Dutch, Hungarian, Hindi, Italian, Telugu, German, and Norwegian Bokmål.

7.22.54 Version 0.34.0 (2018-07-16)

- Prompt for secret during firstboot welcome
  - (Does not apply to downloadable FreedomBox images, but only when installed using freedombox-setup package.)
- Updated translations for Italian, Dutch, Hindi, Hungarian

7.22.55 Version 0.33.1 (2018-07-04)

- Fix issue where editing a user would remove them from admin group
- Updated translations for Hungarian, Czech, Spanish, Russian, Hindi

7.22.56 Version 0.33.0 (2018-07-02)

- Updated translations for Hungarian, Norwegian Bokmål, Spanish, Russian, Czech, Hindi, Dutch, Italian
- firewall: Display information that a service is internal only
- users: Don't show Create User link to non-admin users
- users: Redirect to users list on successful user creation
- packages: Show button to refresh package lists when a package is not available for install
- Only show front page shortcuts that a user is allowed to access
- Restrict removal of last admin user
- Use logos instead of icons in the apps page
- udiskie: New module for automatic mounting of removable media

7.22.57 Version 0.32.0 (2018-06-18)

- Apply new card based design
- Fix client info table size and flickering
- first-setup: Automatically expand root partition
- mediawiki: Enable image uploads
- mediawiki: Make private mode and public registrations mutually exclusive
- mediawiki: Hide frontpage shortcut when private mode is enabled
- Updated translations for Norwegian Bokmål, Czech, Spanish, Russian, Hindi, Telugu, Italian, Dutch, German, and Hungarian

7.22.58 Version 0.31.0 (2018-06-04)

- Updated translations for Czech, Spanish, Russian, German, Italian, Hindi, Telugu, and Norwegian Bokmål
- mediawiki: Added private mode option
- users: Fix user permissions not being saved
- users: internationalize a string
- mediawiki: Run update script for 1.30 upgrade
- shortcuts: Fix urls for ikiwiki shortcuts

7.22.59 Version 0.30.0 (2018-05-21)

- Updated translations for Russian, Italian, Norwegian Bokmål, Hungarian, and Hindi
- setup: Remove unavailable as a state in setup_helper

7.22.60 Version 0.29.1 (2018-05-08)

- security: Fix issue with Plinth locked out from sudo
- Updated translations for Czech and Spanish

7.22.61 Version 0.29.0 (2018-05-07)

- security: Allow console login access to user plinth
- Add an option to enable/disable public registrations in mediawiki
- tt-rss: Skip the check for SELF_URL_PATH
- searx: Fix issue with uwsgi crashing
- Updated translations for Czech, Spanish, German, Norwegian Bokmål, and Italian

7.22.62 Version 0.28.0 (2018-04-23)

- setup: disable install button for currently unavailable apps
- Add locale for Lithuanian (lt)
- Translation updates for Italian, Czech, Russian, Spanish, German, Norwegian Bokmål, Telugu, and Dutch

7.22.63 Version 0.27.0 (2018-04-09)

- middleware: Skip 'installed' message for essential apps
- users: Fix admin group appearing twice in permissions
- apps: Fix app names and short descriptions not being translated
- snapshots: Move manual page link to the index page
- UI: Fix progress bar not appearing
- snapshots: Fix for permissions issue when updating configuration
- snapshots: Add option to enable/disable software installation snapshots
- Translation updates for Italian, Czech, Russian, Spanish, Dutch, German, Norwegian Bokmål, and Ukrainian

7.22.64 Version 0.26.0 (2018-03-26)

- snapshots: Update description
- searx: Rewrite url from /searx to /searx/
- manual: Link to manual from each service
- Workaround security issues in django-axes
- apache: Only regenerate snake oil cert when needed
- apache: Explicitly enable the latest version of PHP module
- apache: Increase module version number to fix php7.2
- Update translations for Chinese (Simplified), Russian, Czech, German, Norwegian Bokmål, Hungarian, Spanish, and Italian

7.22.65 Version 0.25.0 (2018-03-12)

- sharing: Add app for sharing disk folders.
- ttrss: Update list of client apps.
- infinoted: Allow setup to recover after timeout issue.
- snapshots: Add configuration tab with settings for time-based snapshots.

7.22.66 Plinth v0.24.0 (2018-02-26)

- Add file-sharing application Coquelicot.
- Add metasearch engine application Searx.
- Add locale for Hungarian (hu).
- mediawiki: Allow shortcut to be publicly visible on front page.
- clients: Add and correct Client Apps.
- locale: Preferred language can be set in each user's profile.
- locale: Anonymous users can select preferred language.
- config: Remove language selection from config page.
- matrixsynapse: Fix mail attribute for ldap login.

7.22.67 Plinth v0.23.0 (2018-02-12)

- snapshots: Modify configurations to reduce disk usage.
- snapshots: Skip currently active snapshot when deleting all snapshots.
- jsxc: Use consistent url format.
- sso: Increase timeout to 60 minutes.
- theme: Change font from Helvetica to Lato.
- Translation updates for Czech, German, Gujarati, and Telugu.

7.22.68 Plinth v0.22.0 (2018-01-30)

- matrix-synapse: Make sure configuration file does not get corrupted.
- tor: Show enabled status properly.
- first_setup: Fix not showing admin user creation step.
- Migrate from GitHub to Salsa
- Migrate from CircleCI to GitLab CI on Salsa.
- Translation updates for Czech, Dutch, Gujarati, Hindi, Russian and Telugu.
- Started new translation for Ukrainian.

7.22.69 Plinth v0.21.0 (2018-01-15)

- navigation bar: Change label from 'Configuration' to 'System'.
- storage: Removed beta warning for expanding partition.
- groups: Consistently show available user groups, even before applications are installed.
- syncthing: Restrict administration to users in "syncthing" group.
- help: Show menu on smaller screens also.
- diagnostics: Enable the "Run Diagnostics" button when applications are enabled but not running.

7.22.70 Plinth v0.20.0 (2018-01-01)

- bind: Don't use forwarders by default
- ejabberd: Remove redundant button Client Apps
- mediawiki: Add wiki application
- users: Make sure first run actually works
- bind: Add information about current utility

7.22.71 Plinth v0.19.0 (2017-12-18)

- ejabberd: Use dynamic reload instead of restart when changing configuration.
- manual: Make manual available as a PDF download.
- minetest: Show domain information for users to connect to minetest.
- snapshots: Add button to delete all snapshots.
- snapshots: Add option to enable/disable automatic timeline snapshots.
- users: Add groups for bit-torrent and feed-reader, available when these applications are installed.

7.22.72 Plinth v0.18.0 (2017-12-04)

- Add Shadowsocks client with socks5 proxy.
- Fix SSO regressions and conflict with captcha.
- transmission: Fix sso not being enabled on upgrade.
- avahi: Add service for FreedomBox discovery.
- Add client information for modules.

7.22.73 Plinth v0.17.0 (2017-11-20)

- transmission: Enable Single Sign On.
- cockpit: Add short description to frontpage shortcut.
- fail2ban: Fix spelling and sentence structure.

7.22.74 Plinth v0.16.0 (2017-11-06)

7.22.74.1 Added

- Add mobile, web and desktop client info for modules.
- Enable django SecurityMiddleware to improve security ratings.
- cockpit: New module for server administration and web terminal.

7.22.74.2 Fixed

- letsencrypt: Fix internal server error when obtaining a certificate.
- ejabberd: Fix LDAP server entry in config file during setup.
- jsxc: Fix outdated URLs for connecting to local ejabberd server.

7.22.75 Plinth v0.15.3 (2017-10-20)

7.22.75.1 Changed

- Rename Disks to Storage.
- Rename Snapshot to Storage Snapshots.
- tt-rss: Enable API access by default.
- Allow access to Plinth from outside the LAN.
- matrix-synapse: Disable public registration by default.
- power: Merge actions into the user dropdown.

7.22.75.2 Added

- Add locales for Kannada (kn) and for Bengali (bn).
- ejabberd: Use Let's Encrypt certificate, also across renewals.
- matrix-synapse: Add enable/disable public registrations.
- Add captcha validation on 3 failed attempts.
- matrix-synapse: Enable LDAP integration.
- letsencrypt: Automatically obtain and revoke SSL certificates.

7.22.75.3 Fixed

- Fix front page label names.
- Fix vertical alignment of shortcut icons.
- storage: Fix issue with locales that use other decimal separators.
- Make tt-rss api accessible using Apache basic auth.
- letsencrypt: Handle case where current domain is empty.
- Handle both admin and non-admin user names in update user template.

7.22.76 Plinth v0.15.2 (2017-09-24)

7.22.76.1 Added

- letsencrypt: Show more info on cert validity status.
- letsencrypt: Add option to delete certificates.
- letsencrypt: Add option to let Plinth manage certbot's renewal hooks.
- power: Warn if a package manager is running before shutdown/restart.
- security: Install and manage fail2ban.
- names: Include domain and services from dynamicdns.
- disks: Add low disk space warning to system and disks page.
- ssh: New application to manage SSH server.
- Add api module to get enabled services and access info.
- Add Django password validators.
- ejabberd, ikiwiki, ttrss: Add user login descriptions.

7.22.76.2 Removed

- diaspora: Disable for this release due to issues affecting package.
- Remove help from navbar before firstboot complete.

7.22.76.3 Fixed

- i18n: Don't use backslash-newline for wrapping long lines.
- radicale: Update link to documentation.
- sso: Upgrade crypto to 4096-bit RSA and SHA-512.
- Users: Allow non-admin users to log out.

7.22.76.4 Changed

- letsencrypt: Make Let's Encrypt an essential module.
- UI: Make apps and configure pages responsive on small screens.
- Make help accessible for logged-in non-admin users.

7.22.77 Plinth v0.15.0 (2017-07-01)

- Added Tahoe-LAFS module for distributed file storage.
- Added Diaspora* module for federated social networking.
  - Currently only available in "contrib" repository.
- New Locales for Czech (cs) and Tamil (ta).
- Added SSO using auth_pubtkt for Syncthing, TT-RSS, and the Repro admin panel.
  - If you are logged in to Plinth, you will be automatically logged in to these web apps.
- ejabberd: Added option to enable/disable Message Archive Management.
- help: Added Debian release name to about page.
- firstboot: De-bloat first welcome screen.
- Pinned footer to the bottom of the viewport.
- disks: Restrict precision of reported available space on root partition.
- diagnostics: Disable button if app/service is not running.
- help: Only show help pages if user is logged in.
- navbar: Moved logout to user drop-down and added a new power drop-down.
- disks: Show disabled partition resize option if no space is available.
- Added line break to titles to fix frontpage layout.
- syncthing: Fixed typos and clarity in description.
- firewall: Fix 500 error when firewalld is not running.
- setup: Disable install/upgrade when dpkg/apt is running.
- disks: Use information from lsblk for more accuracy.
- datetime: Show timezone properly when it not in expected list.

7.22.78 Plinth v0.14.0 (2017-04)

- tor: Added option to use upstream bridges.
- openvpn: Added shortcut to front page, shown only when logged-in.
- openvpn: Non-admin users can download their own profiles.
- Added new locales for Hindi (hi) and Gujarati (gu).
- Added Syncthing module for file synchronization.
- Added Matrix Synapse as chat server with groups, audio and video.
- Require admin access for all system configuration pages.
- Changed appearance of topbar and footer.
- openvpn: Regenerate user key or certificate if empty.
- disks: Workaround issue in parted during resize.

7.22.79 Plinth v0.13.1 (2017-01-22)

- Two new apps were added:
  - Gobby Server (infinoted) for collaborative editing of text documents
  - Domain Name Server (BIND), in system menu
- Added JavaScript license web labels to provide partial support for LibreJS.
- Added basic configuration form for Minetest server.
- Added indicator to Help->About page if new Plinth version is available.
- Show app logos on front page instead of generic icons.
- Prevent anonymous users from accessing setup pages.
- Split Chat Server (XMPP) app into Chat Server (ejabberd) and Chat Client (jsxc).

7.22.80 Plinth v0.12.0 (2016-12-08)

- Open up RTP ports in the firewall for repro (SIP server).
- Front page shortcuts for services show a Configure button in the details box for logged-in users.
- Add mods packages to be installed with Minetest server.
- Fix issue with reading Dynamic DNS status as non-root user.
- After the hostname is changed, ensure the domain name is still set correctly.
- Allow the domain name to be cleared, and properly set the configuration in this case.
- On the Certificates (Let's Encrypt) page, show a more informative message when no domains are configured.
- On the Chat Server (XMPP) page, show more clearly if domain is not set.
- Apps that require login will not be shown on the front page, unless the user is logged in.
- Show status block for News Feed Reader (Tiny Tiny RSS).
- Change appearance of front page with larger icons and repositioned text.

- Firewall page only lists services that have been setup. The port lists are collapsible under each service.
- Support configuring IPv6 networks.
- Make it less likely to accidentally delete the only Plinth user.
- Updated to work with JSXC 3.0.0 (XMPP web client).

7.22.81 Plinth v0.11.0 (2016-09-29)

- Added loading icon for additional busy operations.
- Added basic front page with shortcuts to web apps, and information about enabled services.
- networks: Add batctl as dependency, required for batman-adv mesh networking.
- users:
  - Fixed checking restricted usernames.
  - Display error message if unable to set SSH keys.
  - Flush nscd cache after user operations to avoid some types of errors.
- monkeysphere:
  - Adopted to using SHA256 fingerprints.
  - Sort items for consistent display.
  - Handle new uid format of gpg2.
  - Fixed handling of unavailable imported domains.
- minetest: Fixed showing status block and diagnostics.
- Fixed stretched favicon.
- Switched base template from container-fluid to container. This will narrow the content area for larger displays.
- Plinth is now able to run as "plinth" user instead of root user.
- xmpp: Replaced jwchat with jsxc.
- ikiwiki: Allow only alphanumerics in wiki/blog name to avoid invalid paths.

7.22.82 Plinth v0.10.0 (2016-08-21)

- Updated Plinth to support Django 1.10.
- Added a page to display recent status log from Plinth. It is accessible from the 500 error page.
- Tor: Added options to toggle relay and bridge relay modes.
- Radicale: Added access rights control.
- Ikiwiki: Updated suggested packages.
- Users and Groups: Fixed editing users without SSH keys.
- Networks: Added basic support for configuring batman-adv mesh networking.
- Networks: Fixed incorrect access for retrieving DNS entries.
- New languages:
  - Persian (50% translated)

  - Indonesian (not started, contributions needed)
- New modules added to Plinth:
  - Disks: Shows free space of mounted partitions, and allows expanding the root partition.
  - Security: Controls login restrictions.
  - Snapshots: Manages Btrfs snapshots.

7.22.83 Version 0.9.4 (2016-06-24)

- Added Polish translation.
- Fixed issue preventing access to Plinth on a non-standard port.
- Dealt with ownCloud removal from Debian. The ownCloud page in Plinth will be hidden if it has not been setup. Otherwise, a warning is shown.
- Fixed issue in Privoxy configuration. Two overlapping listen-addresses were configured, which prevented privoxy service from starting.
- Fixed issue that could allow someone to start a module setup process without being logged in to Plinth.
- Fixed issues with some diagnostic tests that would show false positive results.
- Added check to Diagnostics to skip tests for modules that have not been setup.
- Fixed some username checks that could cause errors when editing the user.
- Added sorting of menu items per locale.
- Moved Dynamic DNS and Pagekite from Applications to System Configuration.
- Allowed setting IP for shared network connections.
- Switched Dreamplug image from "non-free" to "free". This means that we no longer include the non-free firmware for the built-in wifi on Dreamplug.
- Added the "userdir" module for the Apache web server. This allows users in the "admin" group to create a folder called "public_html" under their home folder, and to publicly share files placed in this folder.
- New wiki and manual content licence: *Creative Commons Attribution-ShareAlike 4.0 International* (from June 13rd 2016).
- Switched to using apt-get for module setup in Plinth. This fixes several issues that were seen during package installs.

7.22.84 Version 0.9 (2016-04-24)

- Fixed Wi-Fi AP setup.
- Prevent lockout of users in 'sudo' group after setup is complete.
- Improved setup mechanism for Plinth modules. Allows users to see what a module is useful for, before doing the setup and package install. Also allows essential modules to be setup by default during FreedomBox install.
- Added HTTPS certificates to Monkeysphere page. Reorganized so that multiple domains can be added to a key.
- Added Radicale, a CalDAV and CardDAV server.
- Added Minetest Server, a multiplayer infinite-world block sandbox.
- Added Tiny Tiny RSS, a news feed reader.

7.22.85 Version 0.8 (2016-02-20)

- Added Quassel, an IRC client that stays connected to IRC networks and can synchronize multiple frontends.
- Improved first boot user interface.
- Fixed Transmission RPC whitelist issue.
- Added translations for Turkish, Chinese, and Russian. Fixed and updated translations in other languages.
- Added Monkeysphere, which uses PGP web of trust for SSH host key verification.
- Added Let's Encrypt, to obtain certificates for domains, so that browser certificate warnings can be avoided.
- Added repro, a SIP server for audio and video calls.
- Allow users to set their SSH public keys, so they can login over SSH without a password.

7.22.86 Version 0.7 (2015-12-13)

- Translations! Full translations of the interface in Danish, Dutch, French, German and Norwegian Bokmål, and partial Telugu.
- Support for OLinuXino A20 MICRO and LIME2
- New Plinth applications: OpenVPN, reStore
- Improved first-boot experience
- Many bugfixes and cleanups

7.22.87 Version 0.6 (2015-10-31)

- New supported hardware target: Raspberry Pi 2
- New modules in Plinth:
  - Shaarli: Web application to manage and share bookmarks
  - Date & Time: Configure time zone and NTP service
  - Service Discovery: Configure Avahi service
- Documentation revamp including new user manual and developer guide
- Improved diagnostic tests, available in Plinth
- Avoid unnecessary changes when installing on existing Debian system
- Network configuration supports PPPoE connections
- Debian packages can be download over Tor

7.22.88 Version 0.5 (2015-08-07)

- New targets: CubieTruck, i386, amd64
- New apps in Plinth: Transmission, Dynamic DNS, Mumble, ikiwiki, Deluge, Roundcube, Privoxy
- NetworkManager handles network configuration and can be manipulated through Plinth.
- Software Upgrades (unattended-upgrades) module can upgrade the system, and enable automatic upgrades.
- Plinth is now capable of installing ejabberd, jwchat, and privoxy, so they are not included in image but can be installed when needed.

- User authentication through LDAP for SSH, XMPP (ejabberd), and ikiwiki.
- Unit test suite is automatically run on Plinth upstream. This helps us catch at least some code errors before they are discovered by users!
- New, simpler look for Plinth.
- Performance improvements for Plinth.

7.22.89 Version 0.3 (2015-01-20)

- Tor Bridges: All boxes now act as non-exit Tor bridges, routing traffic for the Tor network.
- Firewall: firewall is on by default and is automatically managed.
- Add BeagleBone support. We now have images for BeagleBone, RaspberryPi, VirtualBox i386/amd64, and DreamPlug.
- Ability to enable and use Tor Hidden Services. Works with Ejabberd/JWChat and ownCloud services.
- Enable Tor obfsproxy with scramblesuit.
- Drop well-known root password (an account with sudo capabilities still exists for now but will be removed soon).
- Switch to unstable as suite of choice for easier development.
- Newer images are built with systemd by default (due to Debian change).
- Install and operate firewall automatically (uses firewalld).
- Major restructuring of Plinth UI using Python3, Django web development framework and Bootstrap3. Code quality is much better and UI is more polished.
- Introduced packaging framework in Plinth UI for on-demand application installation.

7.22.90 Version 0.2 (2014-03-16)

- Support for Raspberry Pi and VirtualBox (x86) in addition to the DreamPlug.
- New Services:
  - Configuration Management UI.
  - Instant Messaging.
  - OwnCloud.
  - dnsmasq.
  - Low-Level Configuration Management.
  - Service Announcement.
  - LDAP Server.
  - LXC Support.
  - Source Packages.
- The privoxy setup is now the default from Debian.

7.22.91 Version 0.1 (2013-02-26)

- First FreedomBox software release (0.1 image, developer release).
- Full hardware support in Debian
- Support for DreamPlug.
- Basic software tools selected as common working environment:
  - User interface system "plinth"
  - Cryptography tools: gpg or "monkeysphere"
  - Box-to-box communication design: Freedom-buddy (uses TOR network)
  - Web cleaning: "privoxy-freedombox".

8 Contributing

From code, design and translation to spreading the word and donation, here are a number of ways to contribute to FreedomBox.

8.1 Quick Links

FreedomBox Developer Manual

Progress calls

TODO page

Donation page

8.2 Welcome to newcomers

As a new contributor, you are more than welcome to introduce yourself to others on the FreedomBox discussion forum, mailing list or on the #freedombox IRC channel. In addition to make useful contacts, you can start reporting bugs and translate (see below) the wiki website and the FreedomBox web interface.

8.3 Development priorities

Upcoming priorities are discussed on an regular basis. You find the progress of the FreedomBox Service (Plinth) with its priorities here: issues board and milestones.

Please check next progress calls to keep yourself on track and meet members of the release team. A TODO page aggregates the complete list of the items to work on for FreedomBox.

8.4 Contributions needed

8.4.1 Add an Application

If you are a developer and wish to see an application available in FreedomBox, you can contribute by adding the application to FreedomBox. See the FreedomBox Developer Manual.

8.4.2 Bugs

List of bugs, feature requests and improvements are tracked on the FreedomBox issue tracker. In addition to that, see list of bugs to help out the Debian package we depend on. Also see the FreedomBox packaging team's dashboard for status of various packages that we use.

8.4.3 Code

If you are a developer, you can contribute code to one of the sub-projects of FreedomBox. Step-by-step process of contributing code to FreedomBox is available.

- FreedomBox Service (Plinth): a web interface to administer the functions of FreedomBox.
- Freedom Maker: a script to build FreedomBox disk images for use on various hardware devices or virtual machines.

You can pickup a task from one of the TODO lists. The individual page project pages contain information availabily of the code, how to build and TODO lists.

8.4.4 Design

8.4.4.1 User Experience Design

If you are a user experience designer, you can help FreedomBox with the following items:

- UI experience for the FreedomBox Service (Plinth) web interface
- Web design for freedombox.org, freedomboxfoundation.org and the wiki pages
- Logo and branding (we currently have an identity manual and logos)
- Possible designs for custom FreedomBox cases on single board computers
- User experience design

8.4.4.2 Technical Design

FreedomBox needs your technical expertise to devise implementation plans for upcoming features. You can contribute to the discussion on various technical design and implementation aspects of FreedomBox. See FreedomBox discussion forum's development category.

8.4.5 Donate

The FreedomBox Foundation is a 501(c)(3) federal nonprofit corporation with recognition from the IRS. FreedomBox project is run by volunteers. You can help the project financially by donating via PayPal, Bitcoin or by mailing a check. Please see the donation page for details on how to donate.

8.4.6 Document: User Manual, Website and Wiki

FreedomBox needs better documentation for users and contributors. FreedomBox manual is prepared by aggregating various pages on the wiki and exporting to various formats. The manual is then used in FreedomBox Service (Plinth) and elsewhere.

If you wish to contribute to the FreedomBox wiki (and consequently the FreedomBox manual), you can create a wiki account and start editing.

For contributing to the website please start a discussion on the FreedomBox discussion forum's development category.

8.4.7 Quality Assurance

- FreedomBox already runs on many platforms and it is not possible for developers to test all possible platforms. If you have one of the supported hardware you can help with testing FreedomBox on the platform.
- When an application is made available on FreedomBox, not all of its functionality is tested in the real world by developer doing the work. Deploying the application and testing it will help ensure high quality applications in FreedomBox.

See the quality assurance page for a basic list of test cases to check for and information on reporting bugs.

8.4.8 Localization

All text visible to users of FreedomBox needs to be localized to various languages. This translation work includes:

- Web Interface for FreedomBox
- FreedomBox documentation
- FreedomBox wiki, website and foundation website.
- Django web framework that FreedomBox uses.
- Individual applications that FreedomBox exposes to users.

You can contribute to the localization effort using the web-based tool at Weblate or directly to the source tree via Salsa.

If you wish to see FreedomBox available for one of your languages, please start a discussion on the FreedomBox discussion forum's development category to work with others translating for that language.

For more information, please visit the FreedomBox translators page.

8.4.9 Spread the Word

Speak to your family, friends, local community or at global conferences about the importance of FreedomBox. To be a successful project we need many more participants, be it users or contributors. Write about your efforts at the talks page and on the wiki.

9 Developer Guide

The FreedomBox Developer Manual provides a step by step tutorial for writing apps for FreedomBox and an API reference. It is available from docs.freedombox.org.

10 Hacking

FreedomBox consists of two main projects:

- FreedomBox Service (Plinth), the web interface
- Freedom Maker, a script to build disk images for various hardware

10.1 FreedomBox Service (Plinth)

FreedomBox Service (Plinth) is a web interface to administer the functions of the FreedomBox.

FreedomBox Service (Plinth) is Free Software under GNU Affero General Public License version 3 or (at your option) a later version.

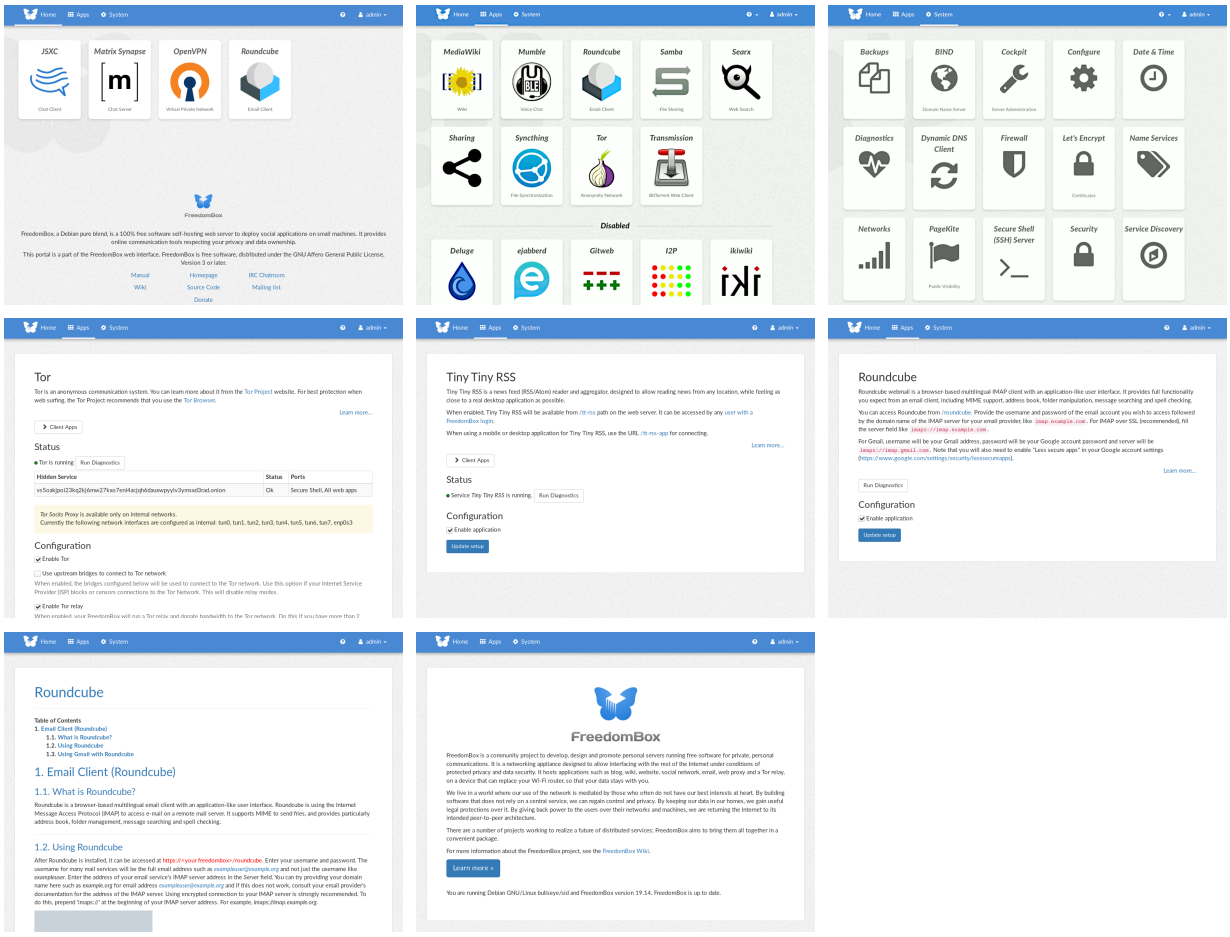
10.1.1 Using

- FreedomBox Service (Plinth) comes installed with all FreedomBox images. You can download FreedomBox images and run on any of the supported hardware. Then, you can access FreedomBox Service (Plinth) by visiting the URL `http://freedombox/plinth` or `https://freedombox.local/plinth`.
- If you are on a Debian box, you may install FreedomBox Service (Plinth) from Debian package archive. Currently, only Stretch (stable), Buster (testing), and Sid (unstable) are supported. To install FreedomBox Service (Plinth) run:

```
$ sudo apt-get install plinth
```

- You can also get FreedomBox Service (Plinth) from its Git repository and install from source.

10.1.2 Screenshots



10.1.3 Support

You may ask for support on

- The discussion forum
- The mailing list
- #freedombox IRC channel
- FreedomBox Matrix channel

10.1.4 Contributing

We are looking for help to improve FreedomBox Service (Plinth). You can contribute to FreedomBox Service (Plinth) by not just by coding but also by translating, documenting, designing, packaging and providing support.

- Instructions on how to contribute code are available.
- The primary Git repository is hosted at FreedomBox Salsa Page.
- Instructions for installing from source and hacking the source are available.
- List of bugs, TODO items and feature requests are available on the issue tracker.
- Before contributing to FreedomBox Service (Plinth) code, you need understand Python and Django on top which it is built.
- You can request for development assistance on the discussion forum, the mailing list or the #freedombox IRC channel.

10.1.4.1 Debian Package

- FreedomBox Service (Plinth) is packaged for Debian. FreedomBox Service (Plinth) is a native package and packaging source code is part of the main package source code.
- Issues related to packaging are listed on Debian BTS.

10.2 Freedom Maker

Freedom Maker is a script to build FreedomBox disk images for use on various hardware devices or virtual machines.

Freedom Maker can currently build FreedomBox disk images for the following:

- A20-OlinuXino-LIME
- A20-OlinuXino-LIME2
- A20-OLinuXino-MICRO
- Banana Pro
- BeagleBone
- Cubieboard2
- Cubietruck
- pcDuino3
- Raspberry Pi 2
- Raspberry Pi 3 Model B
- Raspberry Pi 3 Model B+
- VirtualBox
- QEMU
- AMD64 (x86-64) Machines, X86 Machines and other virtual machines (using raw disk images)

If a hardware platform is capable of running Debian, it should not be too much effort adopt Freedom Maker to create FreedomBox images for the platform.

Freedom Maker is Free Software licensed under GNU General Public License version 3 or (at your option) a later version.

10.2.1 Building FreedomBox Images

- You can get Freedom Maker from its Git repository and follow the instructions in the README to build a FreedomBox image.

10.2.2 Support

You may ask for support on

- The discussion forum
- The mailing list
- #freedombox IRC channel
- FreedomBox Matrix channel

10.2.3 Contributing

We are looking for help to improve Freedom Maker.

- Instructions on how to contribute code are available.
- Freedom Maker is hosted at FreedomBox Salsa Project. The primary Git repository is hosted there.
- You can contribute to FreedomBox by adding support for more hardware platforms. Freedom Maker can be easily adopted to newer platforms if they already support running Debian.
- You can create and test images with Freedom Maker regularly to test for new features and check for regressions.
- List of bugs, TODO items and feature requests are available on the issue tracker.
- You can request for development assistance on the discussion forum, the mailing list or the #freedombox IRC channel.

11 Tell people around you

- FreedomBox
- FreedomBox in the Press
- Conferences
- Talks and presentations
  - Available Material Slides and other raw material
- Facebook
- Twitter
- Mastodon
- Debconf11 Videos