

# WeeNAS



## Installation Guide

**This is an incomplete draft!**

Everything you need to know to get your WeeNAS system up and running.

## **Trademarks, Copyrights, and Disclaimers**

The term Raspberry Pi is a trademark of the Raspberry Pi foundation. <http://www.raspberrypi.org/>

FreeBSD is a registered trademark of the FreeBSD foundation. <https://www.freebsdoundation.org/>

Samba is a trademark of the Samba Project.  
<https://www.samba.org/>

The text of this document is copyright (c)2020 David Horton, and is made available according to the terms of the Creative Commons Attribution Share-Alike (BY-SA) 4.0 license.

<https://creativecommons.org/licenses/by-sa/4.0/>

The source code of the WeeNAS API and its utilities are copyright (c)2020 David Horton and are made available under the conditions of the Simplified BSD license. See License.txt in the root of the WeeNAS software package.

Your data is important to you. You, and only you, are responsible for safeguarding it against loss. Using WeeNAS to store your data in no way removes that responsibility. Always maintain backups and have a recovery plan for when disaster strikes.

## **Installing WeeNAS on FreeBSD 12.1**

This guide will help you get started with WeeNAS by outlining the procedure for installing FreeBSD, the operating system that is the foundation of WeeNAS.

To be successful, you should be familiar with the Raspberry Pi, know the basics of home network configuration, and also how to use Windows-based open-source network utilities. Most of the installation and configuration is scripted, but familiarity with using the command-line is helpful.

If you are an experienced Raspberry Pi tinkerer, you should be fine, but if this is your first RPi project, you may find it easier to start with one of the official Raspberry Pi distributions.

At a high level, these are the tasks:

1. Downloading the FreeBSD operating system image and writing it to an SD card.
2. Booting the Raspberry Pi with FreeBSD and connecting via secure shell.
3. Downloading and unpacking WeeNAS.
4. Running the WeeNAS installer to configure the system as a network attached storage device.
5. Adding users with the WeeNAS web-based administration tool.

## What You Will Need

- A PC or laptop with access to the internet and a MicroSD card slot.
- A 32G Class 10 or faster MicroSD card that is compatible with Raspberry Pi.
- A Raspberry Pi 2B with power supply.
- An internet router with a wired connection for the Raspberry Pi.
- 7-Zip software to uncompress the FreeBSD image.
- Win32DiskImager to write the FreeBSD image to the MicroSD card.
- AngryIP Scanner to find your device's DHCP address.
- PuTTY Secure Shell (SSH) client for initial setup.
- Mozilla Firefox for web-based administration. (Firefox is best supported. Other browsers may work depending on their support for HTML5.)

### Note:

All of the software listed above is open source licensed and costs nothing to use. However, if you find it useful, donations to these projects help keep them going.

## Download FreeBSD

Use an FTP client to visit: <ftp.freebsd.org> or use a web browser and go the HTTP equivalent:

<http://ftp.freebsd.org/>

Browse to the directory for the 12.1 ISO images:  
`/pub/FreeBSD/releases/ISO-IMAGES/12.1`

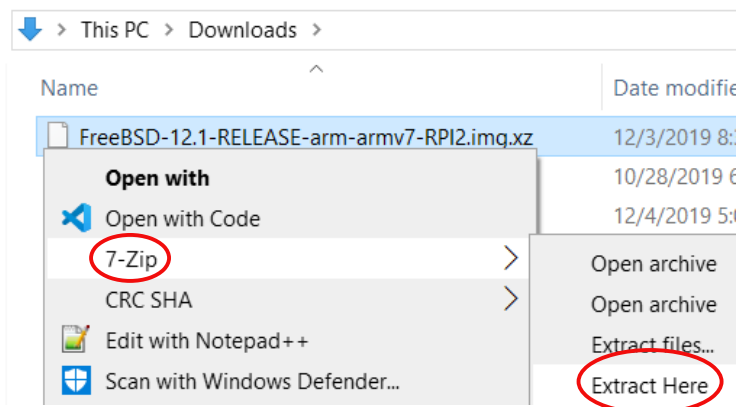
Find the .img.xz for your model of Raspberry Pi.

Remote site:	/pub/FreeBSD/releases/ISO-IMAGES/12.1
Filename	
	FreeBSD-12.1-RELEASE-arm-armv7-PANDABOARD.img.xz
	FreeBSD-12.1-RELEASE-arm-armv7-RPI2.img.xz
	FreeBSD-12.1-RELEASE-arm-armv7-WANDBOARD.img.xz
	FreeBSD-12.1-RELEASE-arm64-aarch64-memstick.img
	FreeBSD-12.1-RELEASE-arm64-aarch64-memstick.img.xz
	FreeBSD-12.1-RELEASE-arm64-aarch64-mini-memstick.img
	FreeBSD-12.1-RELEASE-arm64-aarch64-mini-memstick.img.xz
	FreeBSD-12.1-RELEASE-arm64-aarch64-PINE64-LTS.img.xz
	FreeBSD-12.1-RELEASE-arm64-aarch64-PINE64.img.xz
	FreeBSD-12.1-RELEASE-arm64-aarch64-RPI3.img.xz

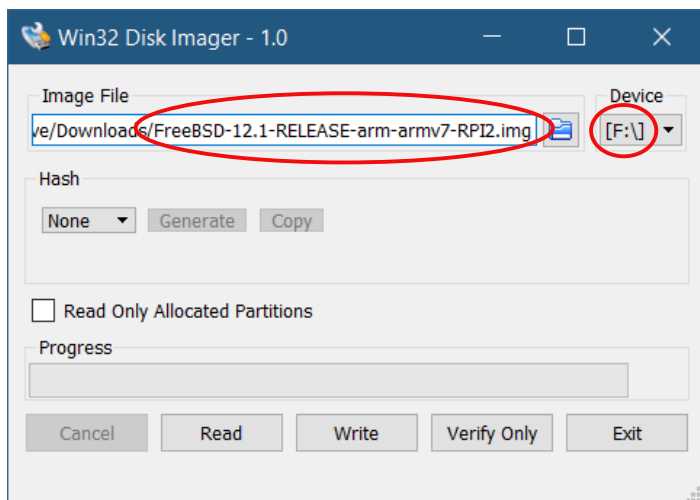
This guide was written using the older Raspberry Pi 2B. Your experience may be different if you have a later revision board.

## Write the Image to the MicroSD Card

First, uncompress the .xz image with 7-Zip by right-clicking the file and using the Extract Here option from the context menu.

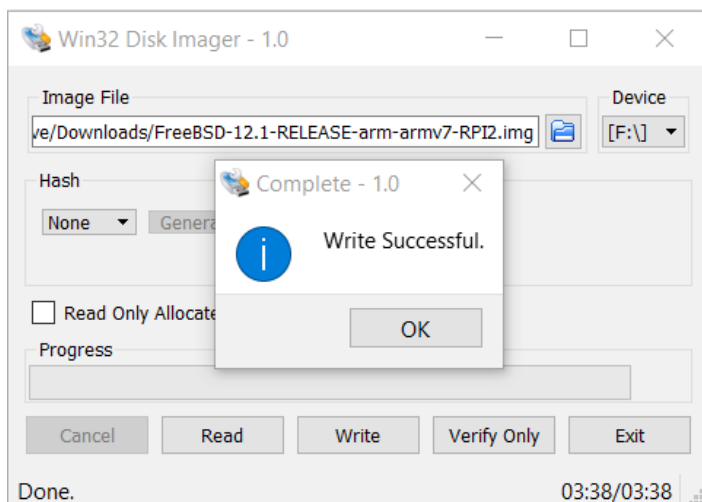


Next, use Win32DiskImager to copy the image onto the MicroSD card.

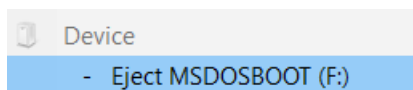


Verify that the correct .img file is selected and verify that the drive letter of the MicroSD card is correct.

If everything looks good, click the Write button to begin. The process takes about four minutes.

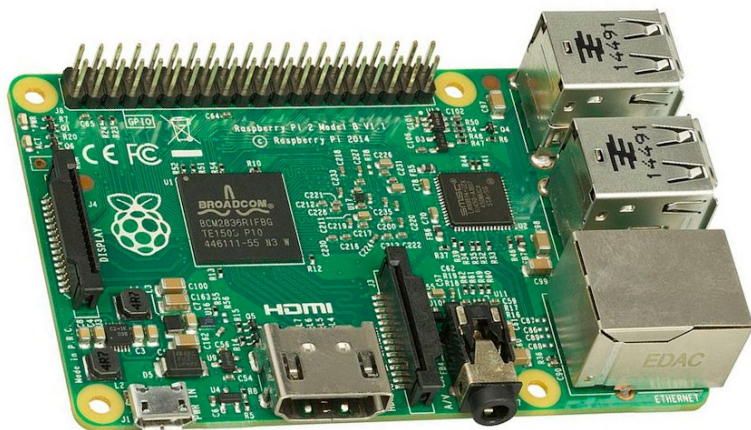


When complete, eject the media and remove the MicroSD card.



## Booting FreeBSD

Insert the MicroSD card into the Raspberry Pi (left side of picture below, on the underside of the board.)



When inserting, remember that the label side of the MicroSD card faces down.

Next, attach an Ethernet cable between the Raspberry Pi's RJ-45 jack (lower right side of the picture) and a port on your internet router.

**No USB devices should be plugged in at this time.**

Finally, plug in the power cable (lower, left corner) and power up the Raspberry Pi.

The initial boot process takes some time and with no monitor attached, it's difficult to see how things are progressing. You can watch the LEDs on the Raspberry Pi to get a rough idea of the status.



After a bit of time, the red LED on the MicroSD socket side of the board will turn off. This means FreeBSD has started booting. Normally, this takes a minute at most. If the LED stays on longer than that, there may be a problem with the image written to the MicroSD card or the card itself.

Next in the boot process, the network link light and traffic indicator LEDs will illuminate, turn off, and come back on. This means the network subsystem is starting and is a good indication that FreeBSD is nearly ready for you to log in. Wait for the link light to remain on for a while before proceeding.

## Finding Your IP Address

Before you can log into FreeBSD on the Raspberry Pi, you have to know the IP address. If your internet router shows a table of connected devices, look for it there under the host name of 'generic'. Otherwise, you can use Angry IP Scanner to find it.

Under the Angry IP Scanner menu, Tools > Fetchers, you can configure the columns of information that will be shown about each device. Add MAC Vendor to the default list.

 IP Range - Angry IP Scanner

Scan Go to Commands Favorites Tools Help

IP Range:	192.168.0.0	to	192.168.0.255	IP Range	▼	⚙
Hostname:		IP↑	Netmask	▼	▶ Start	☰
IP	Ping	Hostname	Ports [12+]	MAC Vendor		

Fetchers

Here you can select fetchers for scanning. Fetchers are represented by columns.

Selected fetchers

Ping  
Hostname  
Ports  
MAC Vendor

↑  
↓  
←  
→  
⚙

Available fetchers

TTL  
Filtered Ports  
Web detect  
HTTP Sender  
Comments  
NetBIOS Info  
MAC Address  
Packet Loss  
HTTP Proxy

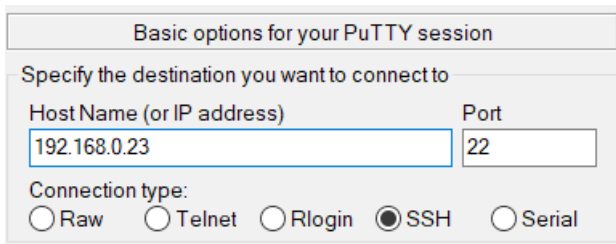
OK Cancel

Run a scan of your network and look for “Raspberry Pi Foundation” or “Raspberry Pi Trading” in the MAC Vendor column. Once you find it, note the IP address listed in the output.

 192.168.0.23	11 ms	[n/a]	22	Raspberry Pi Foundation
--	-------	-------	----	-------------------------

## Logging in via SecureShell (SSH)

Open up PuTTY and enter the IP address you found into the field labeled 'Host Name (or IP address)'.



Basic options for your PuTTY session

Specify the destination you want to connect to

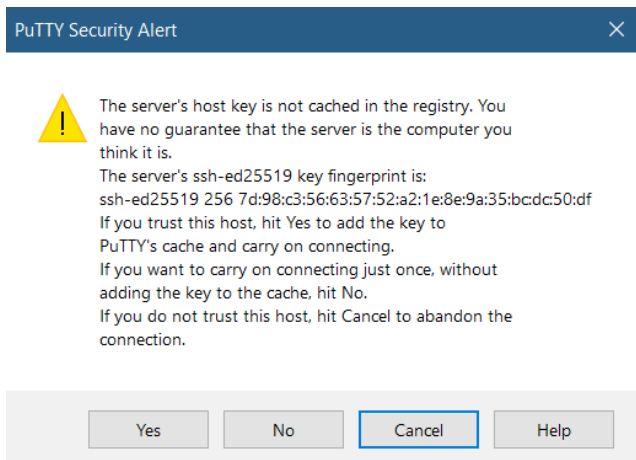
Host Name (or IP address)  Port

Connection type:


☐ Raw ☐ Telnet ☐ Rlogin ☒ SSH ☐ Serial

Click the Open button.

Since this is the first login to this device, you'll get a security alert. You have to choose yes to continue.



PuTTY Security Alert

 The server's host key is not cached in the registry. You have no guarantee that the server is the computer you think it is.

The server's ssh-ed25519 key fingerprint is:  
ssh-ed25519 256 7d:98:c3:56:63:57:52:a2:1e:8e:9a:35:bcdc:50:df

If you trust this host, hit Yes to add the key to PuTTY's cache and carry on connecting.

If you want to carry on connecting just once, without adding the key to the cache, hit No.

If you do not trust this host, hit Cancel to abandon the connection.

After that, a login prompt will appear. Log in with the default username/password combination of freebsd/freebsd.

```
login as: freebsd
Using keyboard-interactive authentication.
Password for freebsd@generic:
```

You'll be treated to some welcome messages and be left at a command prompt. Type 'su -' and enter 'root' when prompted for a password.

```
freebsd@generic:~ % su -
Password:
root@generic:~ #
```

Follow this same procedure any time you need gain superuser access via SSH (Passwords will be different after configuration.)

## Downloading WeeNAS

Download the latest version of WeeNAS using the ‘fetch’ program and the following link:

<https://github.com/DavesCodeMusings/WeeNAS/archive/master.zip>

```
root@generic:~ # fetch --no-verify-peer
https://github.com/DavesCodeMusings/WeeNAS/archive/master.zip
fetch: https://github.com/DavesCodeMusings/WeeNAS/archive/master.zip: size of
remote file is not known
master.zip                                2202 kB 2530 kBps    01s
```

The --no-verify-peer option avoids “Certificate verification failed” errors that crop up.

After downloading, unzip it with ‘unzip master.zip’.

```
root@generic:~ # unzip master.zip
Archive:  master.zip
```

When it’s done, you’ll have a directory named WeeNAS-master.

```
root@generic:~ # ls -F
.cshrc      .login      WeeNAS-master/
.k5login    .profile    master.zip
```

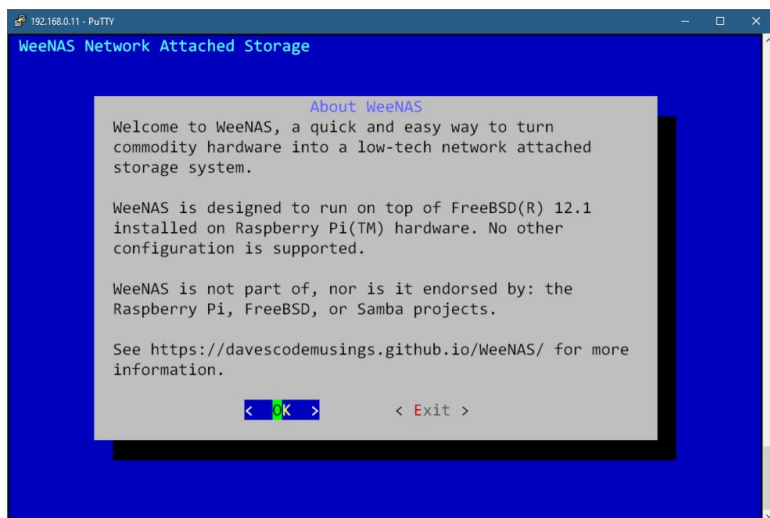
## Installing WeeNAS

The script `install.sh` will guide you through the WeeNAS installation process. The script will detect and configure the USB storage device used for home drives as well as install and configure any binary packages it requires.

First, plug in the USB storage device you want to use for home drives. No other USB flash drives should be plugged in at this time.

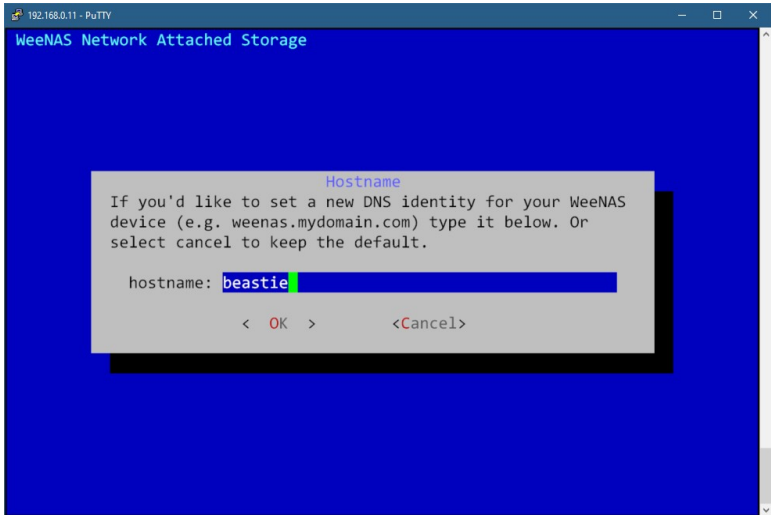
Then, run the script from the `/root/weenas` directory, like this:

```
cd /root/WeeNAS-master
sh ./install.sh
```



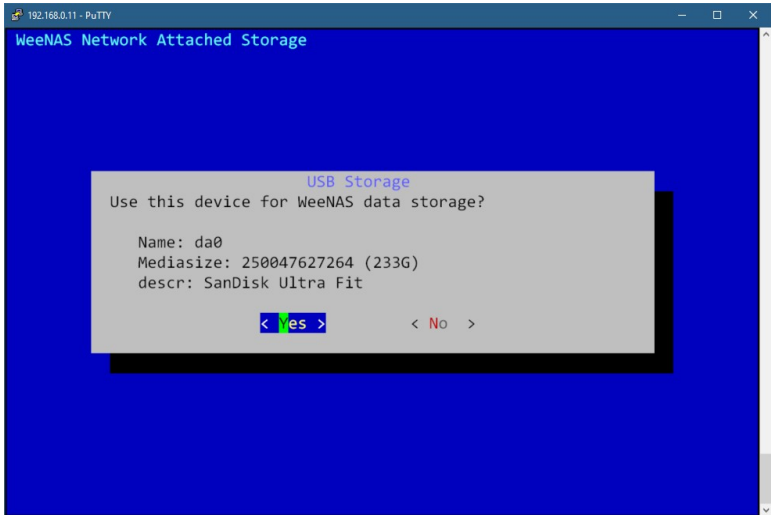
You'll be shown a brief introduction.

You can press Enter or click the mouse on OK to proceed.



You are given the option to name you WeeNAS device with a hostname. If you cancel this, the generic FreeBSD hostname will be used.



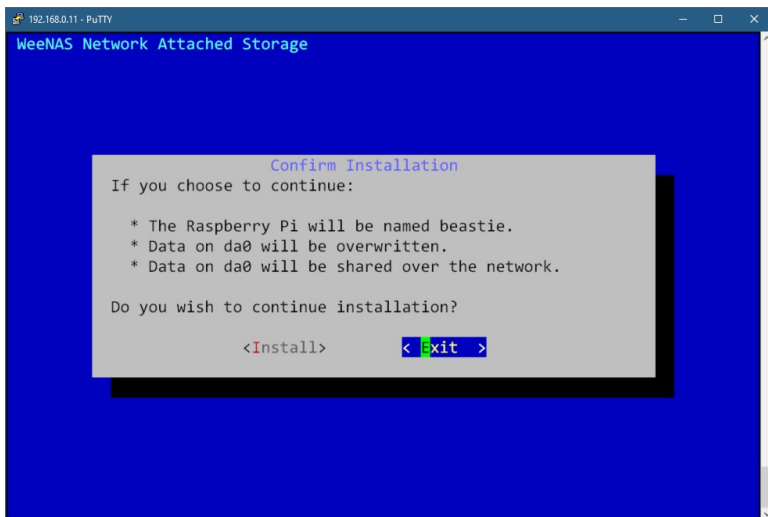


Your USB flash drive will be detected and displayed with information to help you identify it.

If there is more than one USB storage device, you will see the following screen instead.



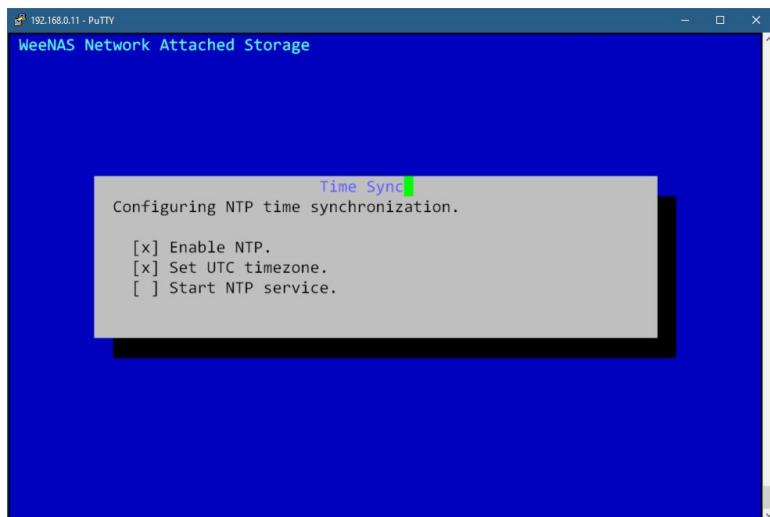
Remove any other devices and choose Rescan. Installation will proceed once the installer detects there is only one flash drive.



You will be asked to confirm installation. This includes overwriting the contents of the USB flash drive if you have directed the installer to do so.

As installation progresses, you will be shown checklists for each task.

For example:



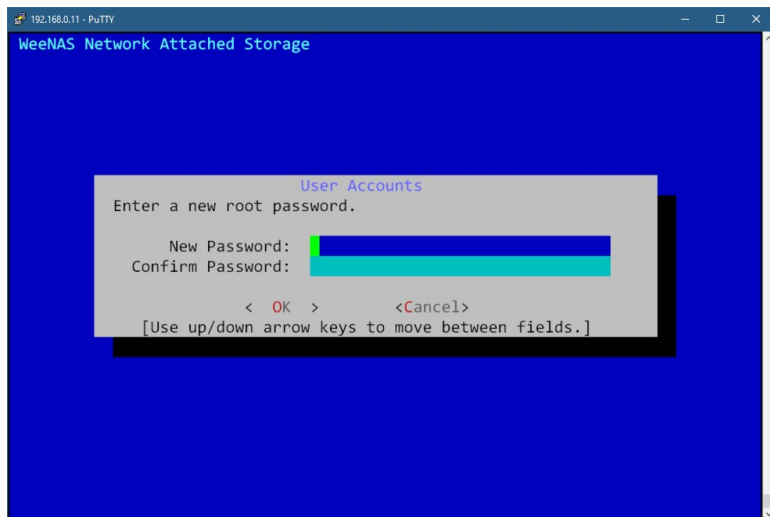
The screenshot above shows tasks being completed for time synchronization. Several informational screens will be shown.

The checklists will update automatically and do not require any interaction. Some of the installation packages are quite large and can take a while to install. Samba, in particular, has over fifty dependent packages to be installed with it and take several minutes to complete.

Note: The installer creates a log file called `/var/log/weenas_install.log`. Open a separate PuTTY session and use the command `tail -f`

/var/log/weenas\_install.log” if you want to monitor detailed progress.

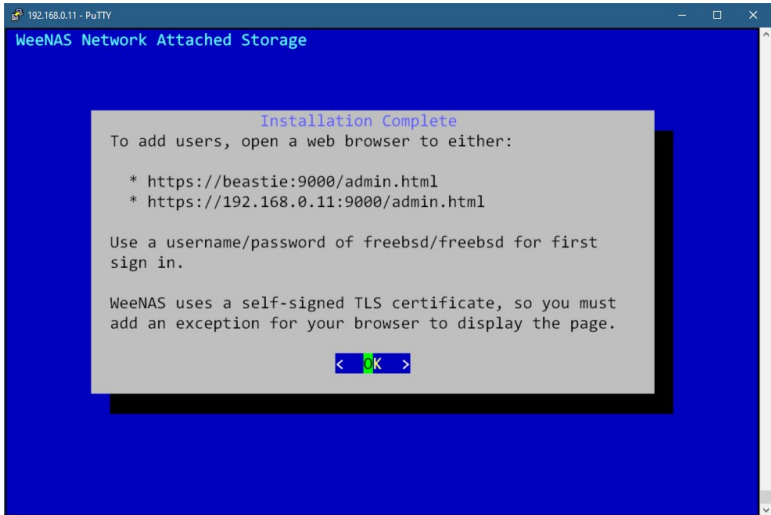
The last item that needs your attention is the root password. The default FreeBSD password is simply ‘root’ and is well documented.



Change the password should to something more challenging or click cancel to keep the FreeBSD default.

Note: Use the arrow keys to move between the fields. Tab will only take you to OK or cancel.

In the end, you will see a message directing you to open a web browser to customize the system.



The image shows a PuTTY terminal window with a blue background. The title bar at the top reads "192.168.0.11 - PuTTY". The main text in the terminal is as follows:

```
WeeNAS Network Attached Storage

Installation Complete

To add users, open a web browser to either:

* https://beastie:9000/admin.html
* https://192.168.0.11:9000/admin.html

Use a username/password of freebsd/freebsd for first
sign in.

WeeNAS uses a self-signed TLS certificate, so you must
add an exception for your browser to display the page.

< OK >
```

When you open the browser, you will get a security warning because of a self-signed encryption certificate.

Here is an example from the Firefox browser:



## Warning: Potential Security Risk Ahead

Firefox detected a potential security threat and did not continue to beastie. If you visit this site, attackers could try to steal information like your passwords, emails, or credit card details.

[Learn more...](#)

[Go Back \(Recommended\)](#)

[Advanced...](#)

beastie:9000 uses an invalid security certificate.

The certificate is not trusted because it is self-signed.

Error code: [MOZILLA\\_PKIX\\_ERROR\\_SELF\\_SIGNED\\_CERT](#)

[View Certificate](#)

[Go Back \(Recommended\)](#)

[Accept the Risk and Continue](#)

Despite the ominous warning, it is safe to proceed. All data will still be encrypted.

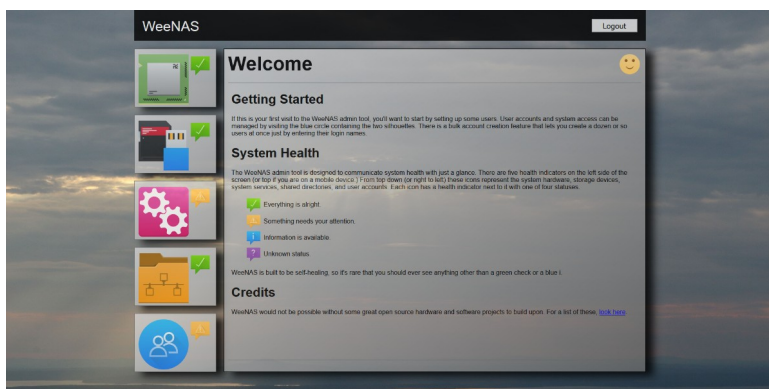
For a more thorough explanation see this article:

[https://en.wikipedia.org/wiki/Self-signed\\_certificate](https://en.wikipedia.org/wiki/Self-signed_certificate)

After the certificate is accepted, the admin login page will appear. It looks like this:



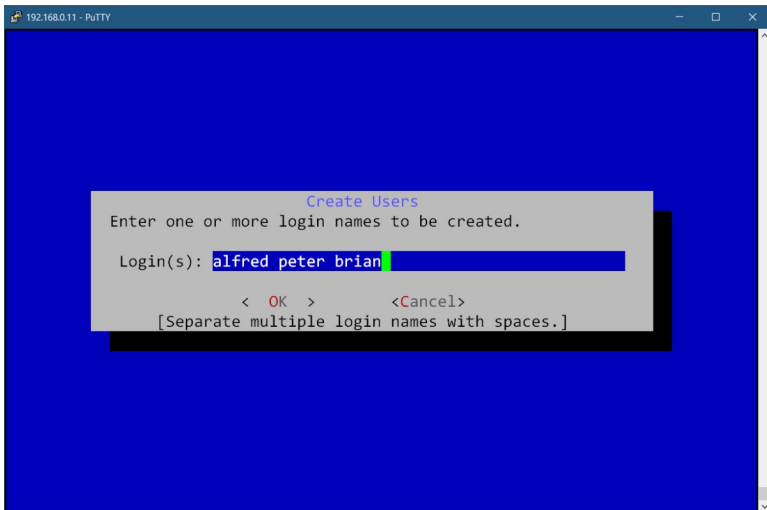
You'll need to log in with the username/password of `freebsd/freebsd` to access the admin page.



Use the admin page to add and remove users\* and view system health and details.

\*Note: Creating users through the admin page is not completed. In the interim, there is a command-line tool called `usercreate.sh`.

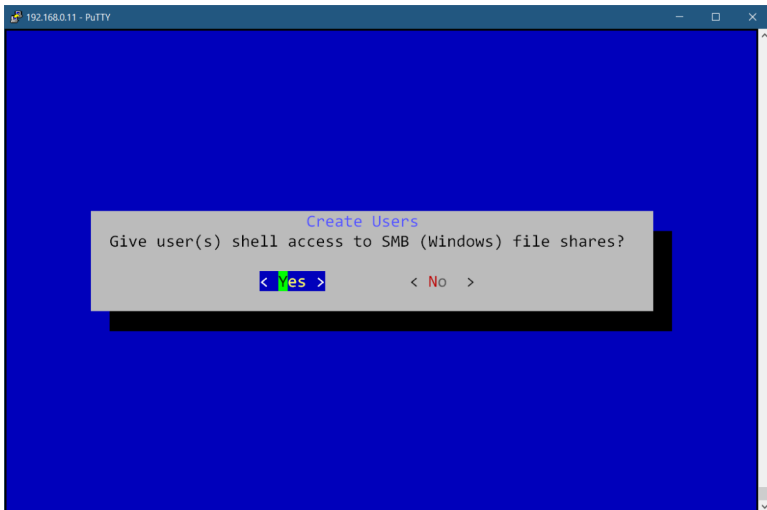
The tool has a look similar to the installer.



First, you will be prompted to enter one or more user account logins. Use lowercase names and separate the names with spaces.

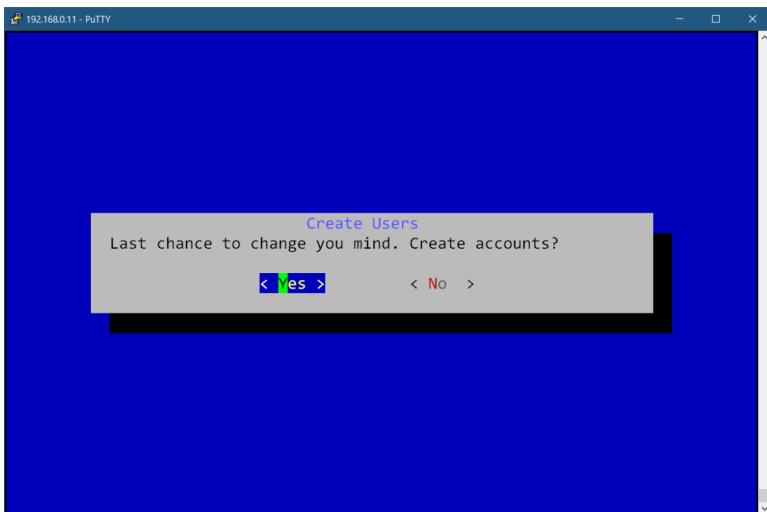
Next, you will be asked a series of questions about access levels. The choices you make will be applied to all users.





SMB file shares and shell access have a default value of yes. Admin access defaults to no.

You will be given one final chance to bail out before users are created.

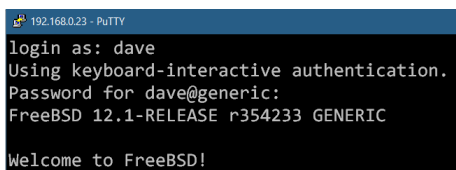


User accounts are assigned a random password.  
Passwords are not recorded.

## Testing Connectivity (SSH)

Secure Shell is your way to access the underlying FreeBSD operating system. This test is very important, because if it does not succeed, you will have no way to access the system.

Open another PuTTY window and enter the IP address of your WeeNAS Raspberry Pi. Log in with the trusted user account and the password given to the 'passwd' command.

A screenshot of a PuTTY terminal window. The title bar at the top reads "192.168.0.23 - PuTTY". The terminal text shows a login process: "login as: dave", "Using keyboard-interactive authentication.", "Password for dave@generic:", "FreeBSD 12.1-RELEASE r354233 GENERIC", and "Welcome to FreeBSD!".

```
192.168.0.23 - PuTTY
login as: dave
Using keyboard-interactive authentication.
Password for dave@generic:
FreeBSD 12.1-RELEASE r354233 GENERIC
Welcome to FreeBSD!
```

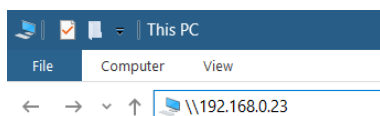
Enter the command 'su -' to switch to the root user.

If you can do all that, your SSH access is fine. Log out and continue with testing access from Windows.

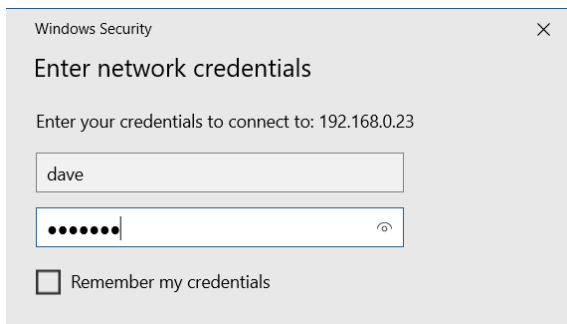
## Testing Connectivity (SMB)

SMB is the Windows way of connecting to network shares. This is how you will access the files stored on the WeeNAS system.

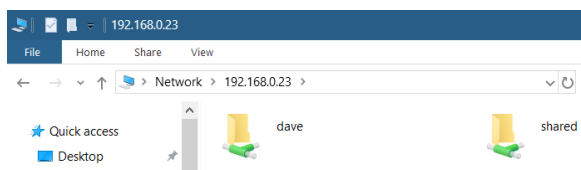
Open up Windows Explorer and enter the IP address of the WeeNAS server preceded by a double backslash.



You should be prompted for a username and password. Enter the trusted username and the password given to the 'smbpasswd' command. You do not need to check the box labeled 'Remember my credentials' at this time.



If all goes well, you should see a network folder with your user account name and possibly a shared folder if you elected to include that in the configuration.



## **Finish**

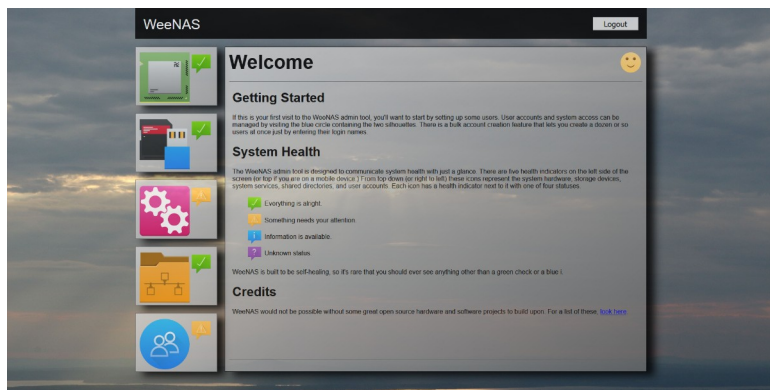
Congratulations! You've installed, configured and tested the installation of your WeeNAS system. The last thing to do is reboot. It's a good way to find out if there are any problems with the system before you start storing files on it.

When you're ready, you can reboot by logging in as a trusted user, 'su -' to root, and then enter the command 'shutdown -r now'.

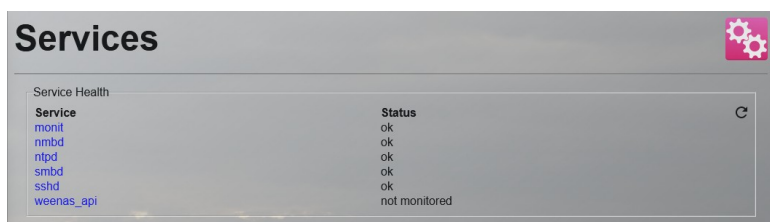
## Troubleshooting

Yellow triangles on the admin page.

The screenshot below shows Services and User Account both with a yellow attention symbol next to them.



To find out what's wrong, click on the left-hand side icon with the attention symbol. In this example, let's start with Services.



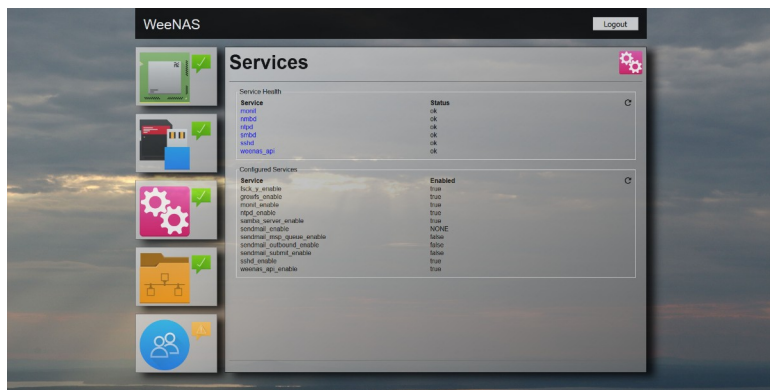
Here, the Service Health section shows the weenas\_api as not monitored. This status means that the Monit program detected a problem, tried to fix it, but failed for some reason.

The first thing to try is to start monitoring again. To do this, log into the system via PuTTY and su - to root.

Tell Monit to start monitoring again by issuing the following command:

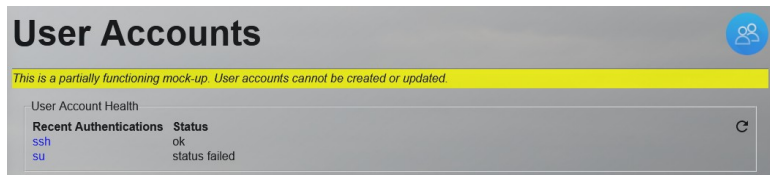
```
root@generic:~ # monit monitor weenas_api
```

Click the browser back button or click the Services icon on the left to return to Service Health.



The weenas\_api status has returned to “ok” and the status indicator on the left will turn to green again after a short delay.

The next problem that needs attention in this example is the User Account section. Click the icon on the left to show details of User Accounts.



Notice how su shows a status of failed.



The text “su” appears as in blue indicating it’s a link you can follow for more information. Clicking the link in this example shows detailed information about the monitoring of su failures. Specifically, there is one failed attempt as shown by the line labeled “last output”.

User Accounts	
Health Details failed_su	
Indicator	Status
data collected	mon, 17 aug 2020 05:16:01
last exit value	1
last output	failed attempts: 1
monitoring mode	active
monitoring status	monitored
on reboot	start
status	status failed

The failed ssh login and failed su attempts are found by examining `/var/log/messages`. It will remain in a failed state until the log file rotates to a new one.

If the failed logins continue to rise, chance are good someone has either forgotten their password, or they are trying to access something they shouldn't.