AEGEUS

Wallet installation guide for staking on a Raspberry Pi

Simple step by step guide

Contents

1 Version History 3
2 Introduction 4
3 Prerequisites 5
4 Installation steps 6
5 Add an address or transfer funds 12
6 Conclusions 19
7 Addendum A - Add an existing address to your wallet 20
8 Addendum B - Control your masternodes 22

1 Version History

Version	Date	Author	Remarks
1.0	August 5th, 2018	tom_d#9324	Initial document
1.1	August 5th, 2018	tom_d#9324	Small corrections, changed listaccounts command to listaddressgroupings
1.2	August 8th, 2018	tom_d#9324	Fixed some bugs and gave a bit more explanation after user testing

2 Introduction

This guide will explain step-by-step how to use the ARM build of the Aegeus wallet to install and configure staking on a Raspberry Pi using the command line interface (CLI).

Note: This guide was made on a Raspberry Pi running the full Raspbian OS, but it should probably work on a Ubuntu 16.04 image also.

Note2: Other ARM-based systems like the Rock64 may also be able to use these binaries (this has yet to be tested)

Why do we want to use CLI instead of GUI (Graphical-user-interface)?

The GUI wallet is much more power and memory hungry compared to the CLI version. So next to drawing more power, it will probably be more unstable and may crash from time to time. I want this Pi to be as stable and power-friendly as possible because it will run 24/7 for staking.

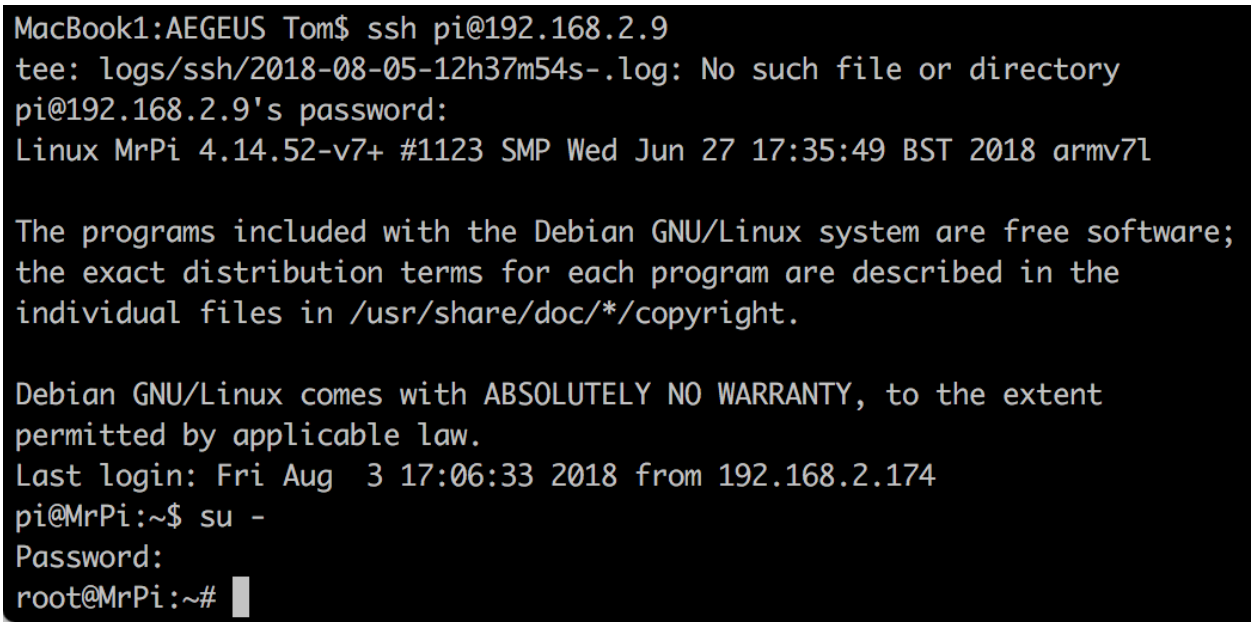
3 Prerequisites

- Raspberry Pi
- Raspbian image burned on SD card
- Network configured (WIFI or hard wired (preferred))
- SSH enabled

4 Installation steps

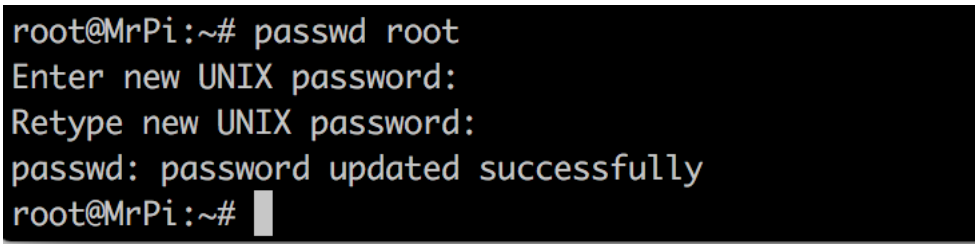
1. Open an SSH session to your Pi and become root (Use your favorite SSH tool for this like terminal, Putty, Bitvise, …). The standard user for a Raspberry Pi running Raspbian is 'pi'
   Note: Optionally you can do this in the terminal on the Pi directly when using a screen/keyboard

```
ssh pi@<IP address>
su -
```



2. If the password for the root user is unknown to you, you will probably still need to set it like this:

```
sudo passwd root
```



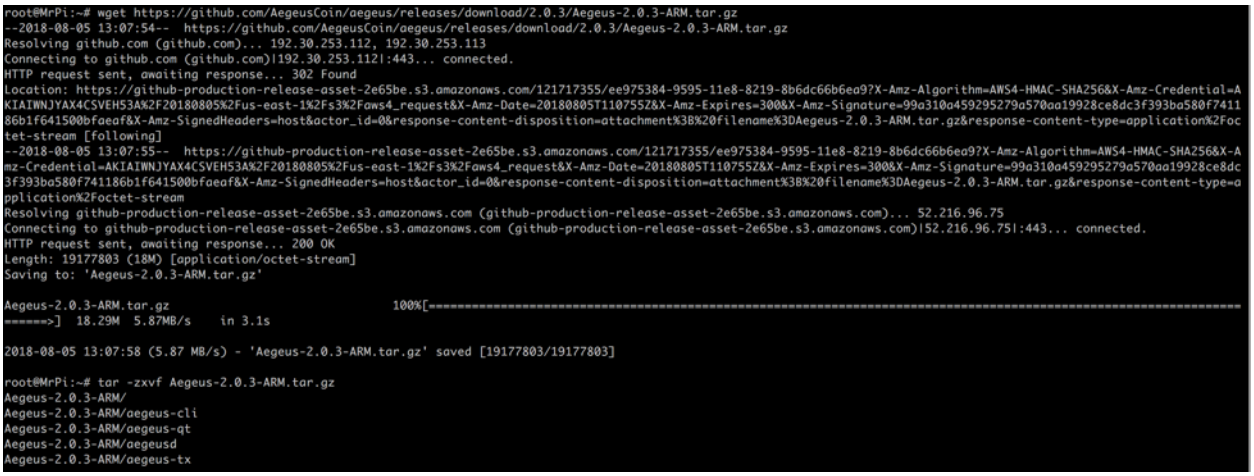
3. Create a working directory

```
cd /root/
mkdir aegeus
cd aegeus
```

4. Download and extract the wallet

   Go to https://github.com/AegeusCoin/aegeus/releases and find the latest ARM build (for example **Aegeus-2.0.3-ARM.tar.gz**)
   Download this build to your Pi with the following command and extract it:

```
wget https://github.com/AegeusCoin/aegeus/releases/download/2.0.3/Aegeus-2.0.3-ARM.tar.gz
tar -zxvf Aegeus-2.0.3-ARM.tar.gz
```



5. Copy the wallet files to /usr/local/bin to make them runnable from everywhere:

```
cd Aegeus-2.0.3-ARM
cp * /usr/local/bin
```

6. Create the aegeus data directory

```
mkdir /root/.aegeus
cd /root/.aegeus
```

7. Install the bootstrap to make the blockchain sync go (very) fast

```
wget
https://gateway.ipfs.io/ipfs/QmUBhH9R5VzrqeiTSia7Sc4HXQTvR9dWgzMDbf8Wi1XTfi
```

Note: If the download is slow, you can choose another IPFS gateway from https://ipfs.github.io/public-gateway-checker/. Just replace the 'https://gateway.ipfs.io/ipfs/' part with another from the list and leave the 'QmUBhH9R5VzrqeiTSia7Sc4HXQTvR9dWgzMDbf8Wi1XTfi' hash in place. Go down the list until you find a fast one.
Extract the bootstrap

```
tar -xvf QmUBhH9R5VzrqeiTSia7Sc4HXQTvR9dWgzMDbf8Wi1XTfi
```

8. Configure aegeus

```
nano /root/.aegeus/aegeus.conf
```

Add the following lines to the file. For the X's, press 16+ random keys on the keyboard. You don't need to record or remember these random values:

```
staking=1
rpcuser=XXXXXXXXXXXX
rpcpassword=XXXXXXXXXXXXXXXXXXXXXX
```

To leave the nano text editor, type the following

Ctrl+x

Yes

Enter

9. Configure the aegeus service

```
nano /etc/systemd/system/Aegeus.service
```

Add the following lines to the file:

```
[Unit]
```

```
Description=Aegeus service
After=network.target
[Service]
User=root
Group=root
Type=forking
#PIDFile=/root/.aegeus/aegeus.pid
ExecStart=/usr/local/bin/aegeusd -daemon -conf=/root/.aegeus/aegeus.conf -
datadir=/root/.aegeus
ExecStop=-/usr/local/bin/aegeus-cli -conf=/root/.aegeus/aegeus.conf -
datadir=/root/.aegeus stop
Restart=always
PrivateTmp=true
TimeoutStopSec=60s
TimeoutStartSec=10s
StartLimitInterval=120s
StartLimitBurst=5
[Install]
WantedBy=multi-user.target
```

To leave the nano text editor, type the following

Ctrl+x

Yes

Enter

10. Configure the aegeus service to start when the system starts

```
systemctl daemon-reload
systemctl enable Aegeus.service
```

11. Cleanup after install

```
rm /root/.aegeus/QmUBhH9R5VzrqeiTSia7Sc4HXQTvR9dWgzMDbf8Wi1XTfi
rm -rf /root/aegeus
```

12. Start the aegeus wallet for the first time.

```
aegeusd -daemon
```

13. Waiting for the blockchain to sync …
    Be impatient and spam getinfo to see if it's ready yet so you can get staking. "Blockchain information not yet available" or "Loading block index" just means it's still starting, which takes a few minutes and quite a bit longer on a single board computer.

```
aegeus-cli getinfo
```

14. Encrypt your wallet
    Remember to be safe and always encrypt your wallet before holding any Aegeus (Do not lose this password as you need it to unlock the wallet or access your funds)
    You can give your wallet a password by using the following command:

```
aegeus-cli encryptwallet <mysupercomplexpasswordhere>
```

    The wallet will turn off once it has completed encrypting. All you need to do is start it again

```
systemctl start Aegeus.service
```

    Note that the systemctl start command does not give any output. You can check if the service was successfully started with systemctl status Aegeus.service or just check aegeus-cli getinfo.

```
systemctl status Aegeus.service
aegeus-cli getinfo
```

    We prefer using ‘systemctl start’ over ‘aegeusd -daemon’ to start the service because systemctl has built in error routines. It will automatically restart the service when it crashes for example.

15. Unlock the wallet
    Now to be able to stake coins you need to unlock your wallet for staking only.
    Note: when you don’t want to store your password in the command history, you can put a ‘space’ before your command

```
walletpassphrase <passphrase> <unlock time> <for  staking/anonymization only
true/false>
```

<passphrase> is the passphrase you created in the previous step

<unlock time> this is the time the wallet will stay unlocked. Use 0 to enable it forever (or better: until the service restarts)

<for staking/anonymization only true/false> If you only want to stake, use true here

for example:

```
aegeus-cli walletpassphrase 6zArYuNpWUrDV2F3N3hdyA 0 true
```

Now the coins in your wallet are ready for staking. How to put funds in this wallet is explained in the next chapter.

5 Add an address or transfer funds

To be able to stake, off course you need to have funds in your wallet. There is more than one way to do this: You can create a new address and transfer funds (this is the preferred way) or you can add an existing address private key.

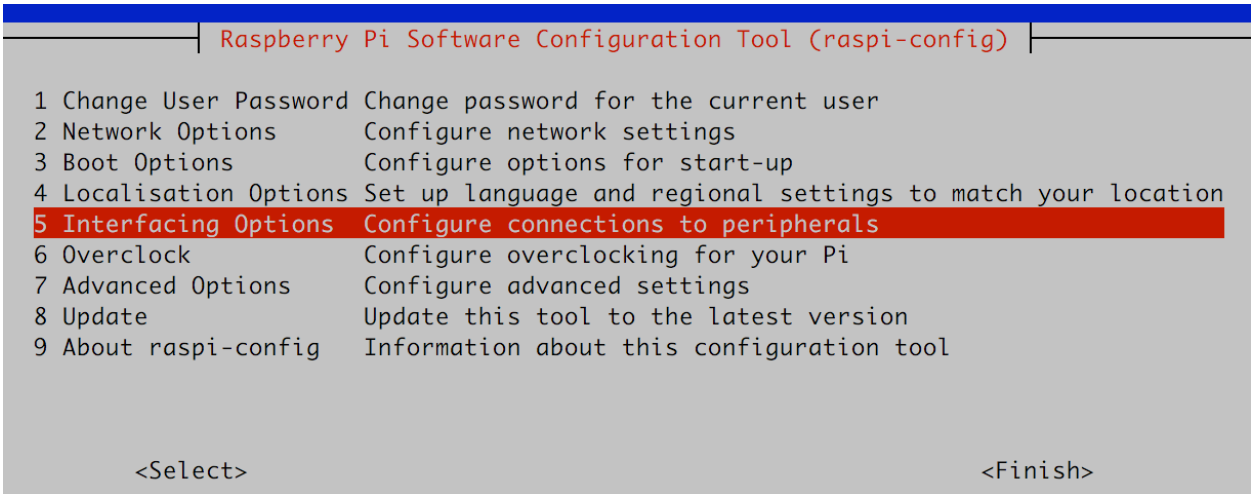
I performed this step in the graphical user interface (I run the full Raspbian with desktop), but you can also do this with the Command line interface (CLI).

1. Because we are going to use the Aegeus graphical wallet, we need to stop the CLI service beforehand (they can't run together)

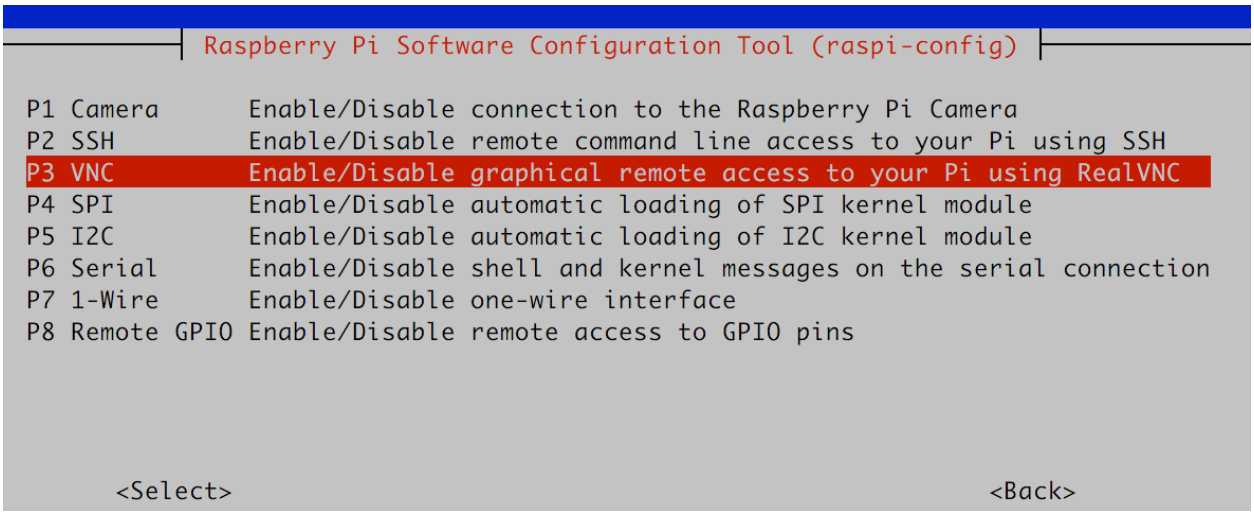
```
systemctl stop Aegeus.service
```

2. Activate VNC server on the Raspbian desktop to be able to remote control the graphical user interface (if you connect a keyboard, mouse and screen, this can also be done locally off course)
Note: If working directly on the Pi desktop, you can skip to step 5, but do a logoff and log on with the root user then.

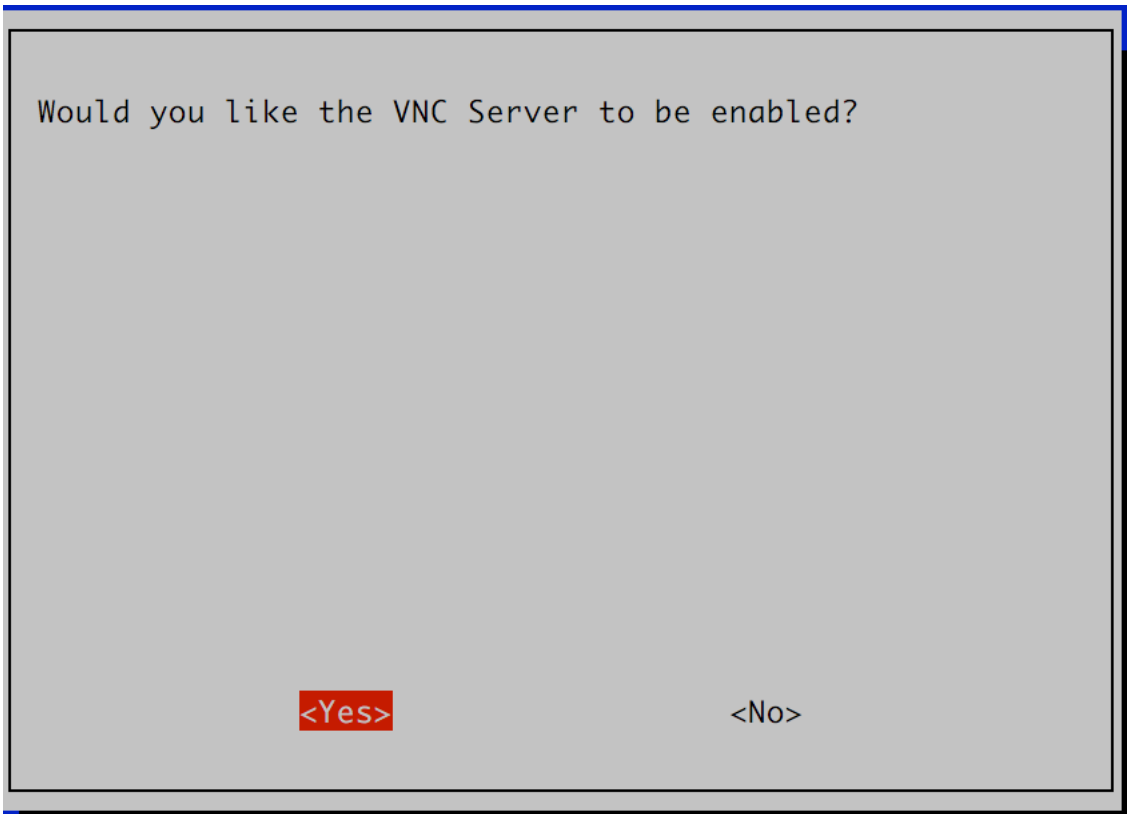
```
raspi-config
```



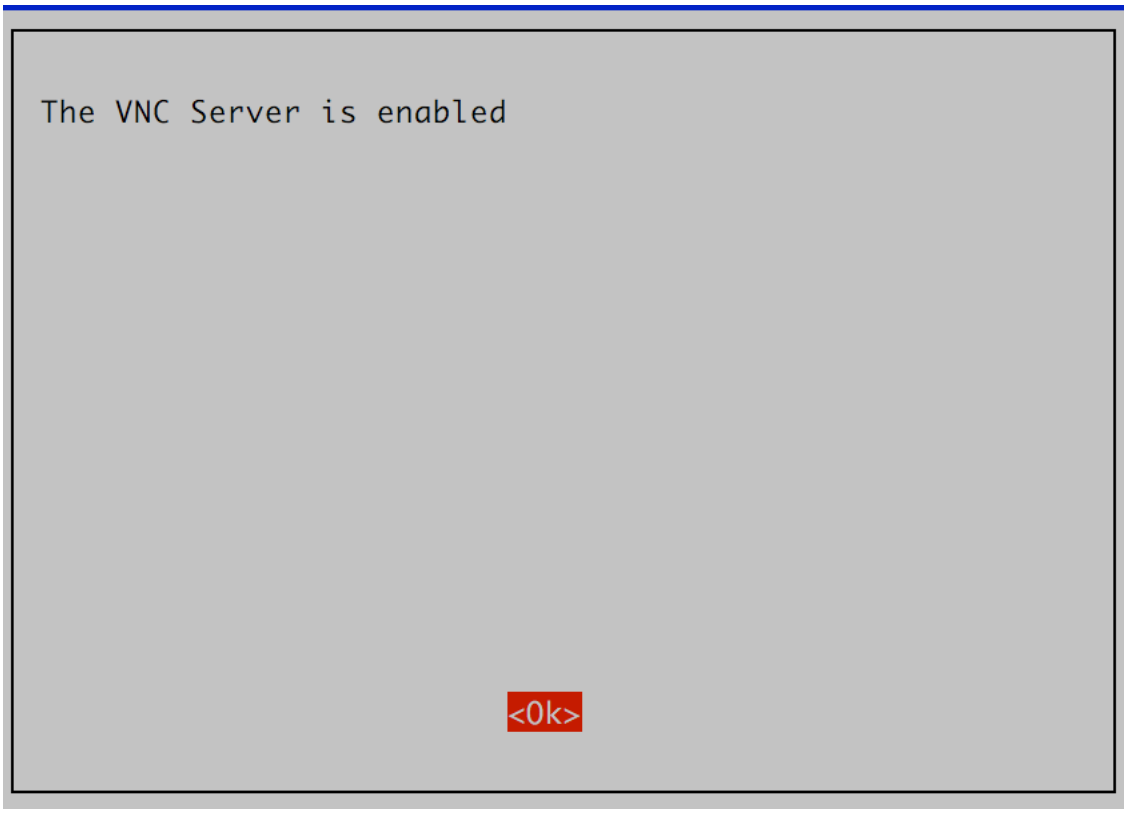
The Raspberry Pi configuration tool will open. Choose 5. Interfacing Options



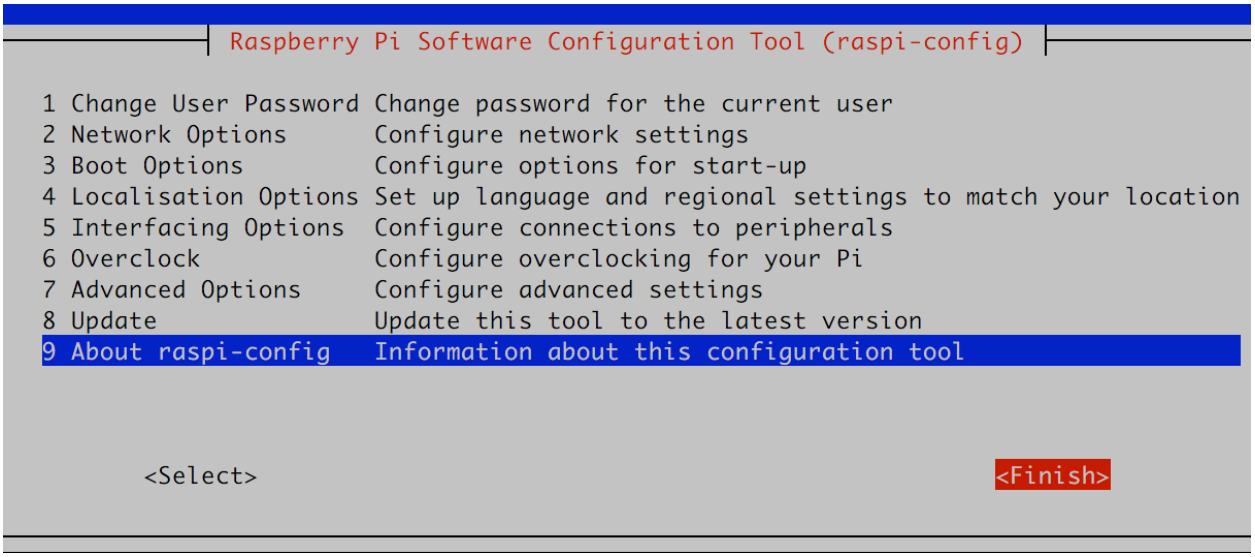
Now choose 3. VNC



Choose Yes



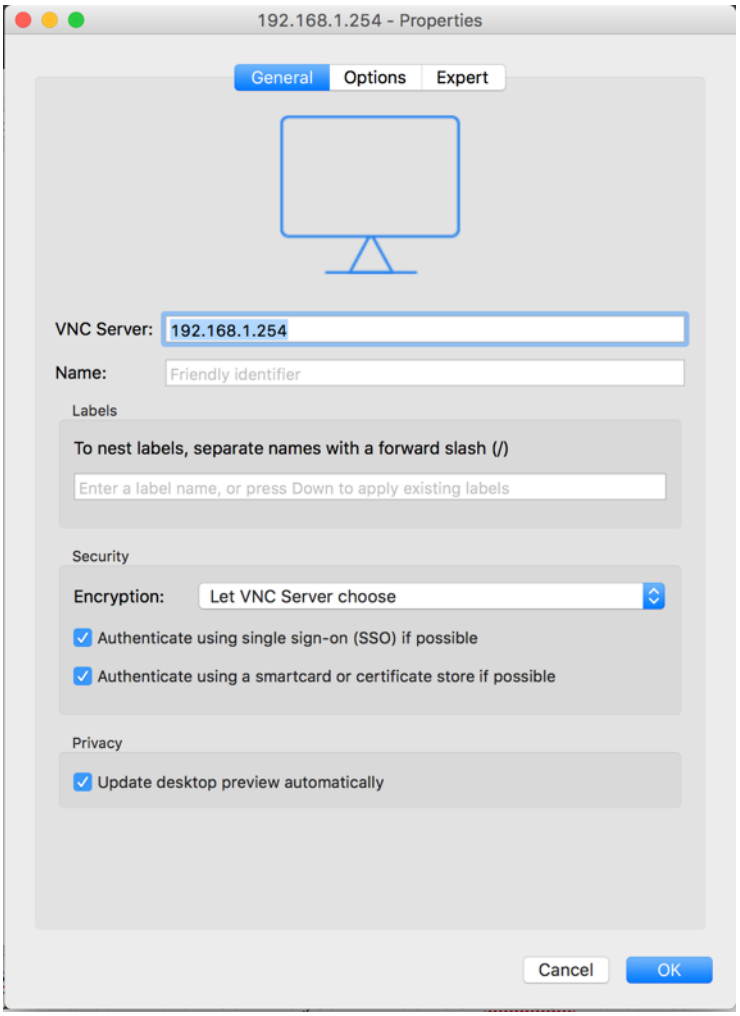
Click OK



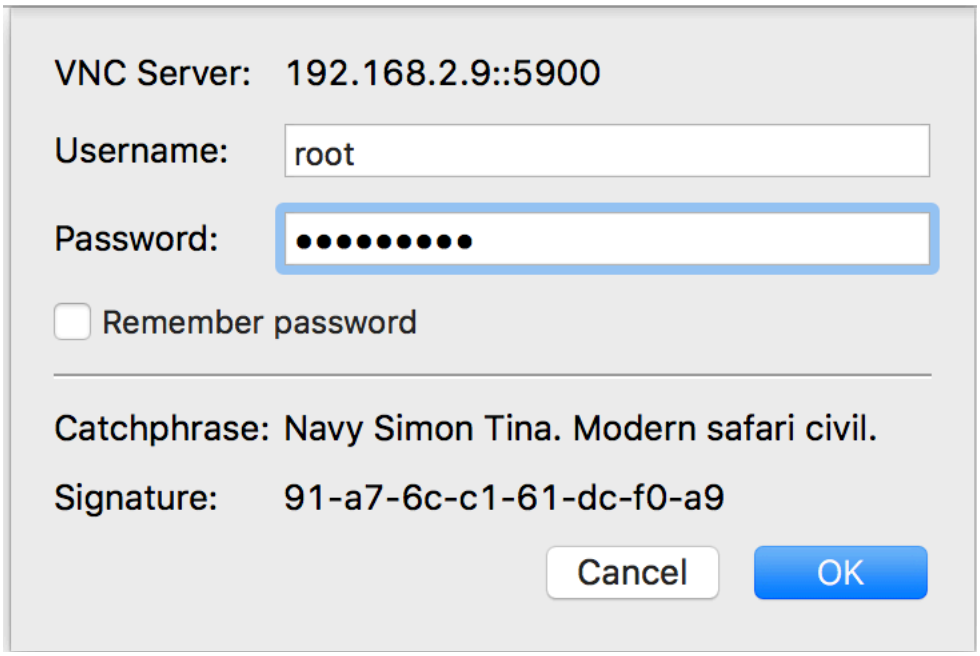
Exit the tool with Finish.

3. Download a VNC viewer (free remote desktop tool) for your Operating System. For example: https://www.realvnc.com/en/connect/download/viewer/windows/

4. Open the VNC viewer and create a new connection to your Pi

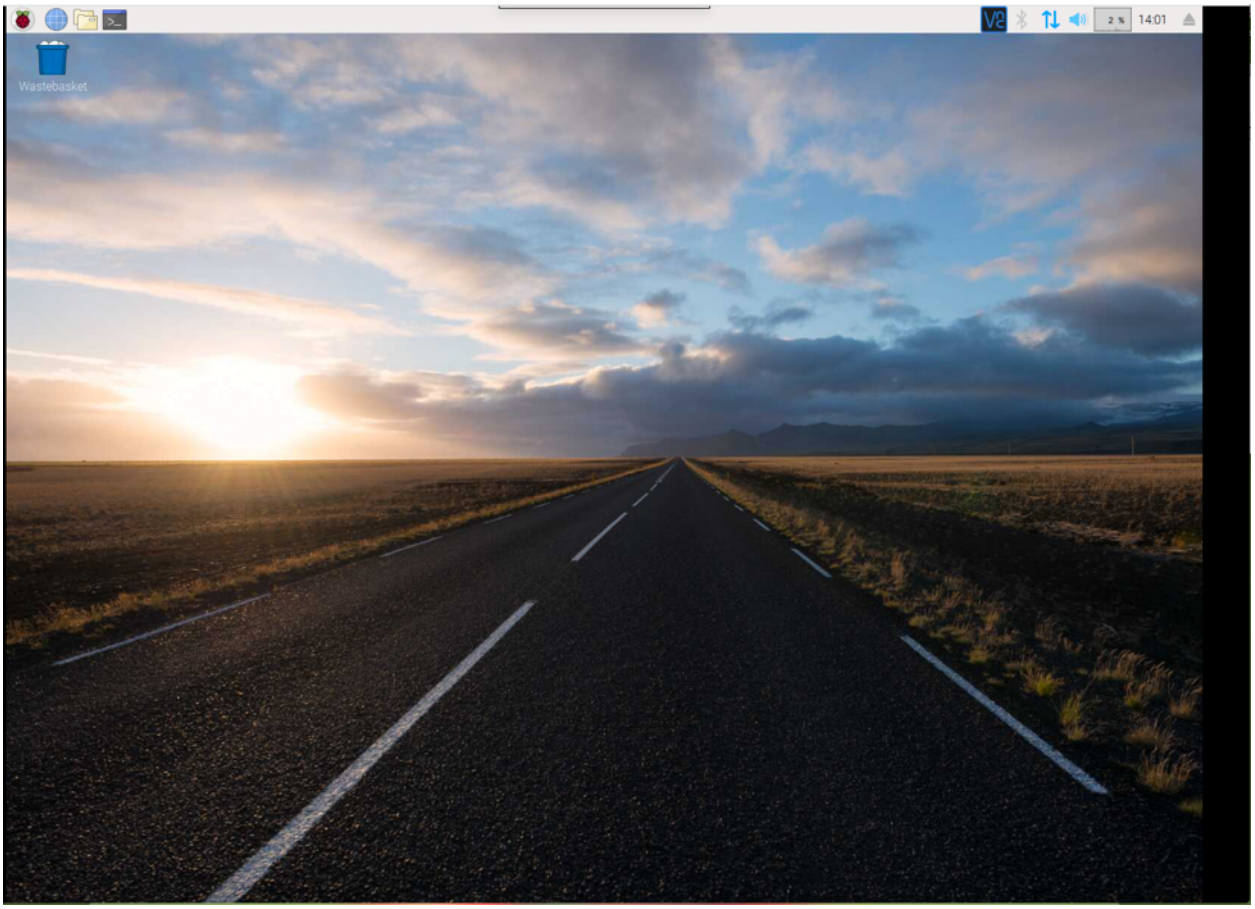


Put in the IP address of you Pi and push Enter.
Now open this connection. The first time it will ask for your password.



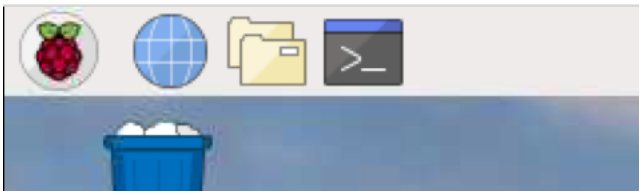
Use the root user and put in your password.

Your Pi desktop will now open

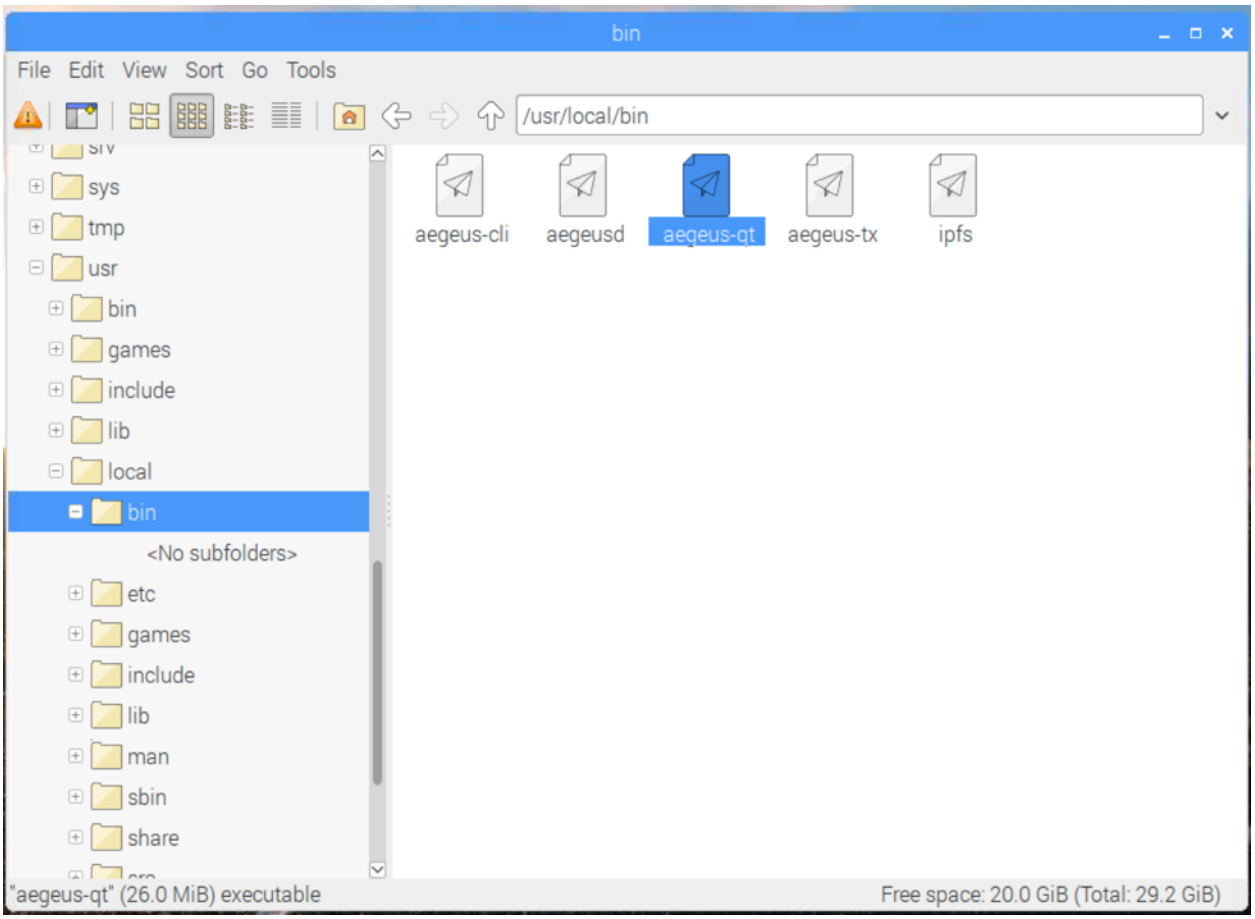


5. Start the graphical Aegeus wallet with the following steps:

Open File Manager in the top left of the screen (yellow icon)



Navigate to /usr/local/bin and open aegeus-qt



This wallet will now look exactly the same as the windows or Mac OSX wallet you are used to.
Wait for the wallet to start up and sync.
You can now unlock your wallet and create addresses , transfer funds, ...
Note: If you want to add an already existing address, please see Addendum A. This is helpful when moving your wallet from another computer to your Pi for example.

6. As soon as you are ready with the wallet, we have to close it to start it up in command-line mode again because this is more stable.
Close the wallet and wait for it to completely close.

Go back to you SSH session and start the Aegeus service again

```
systemctl start Aegeus.service
```

Note: If you opened a new SSH session, become root first with 'su -' Wait for the service to start up (few seconds to a few minutes) and then check your coin balance

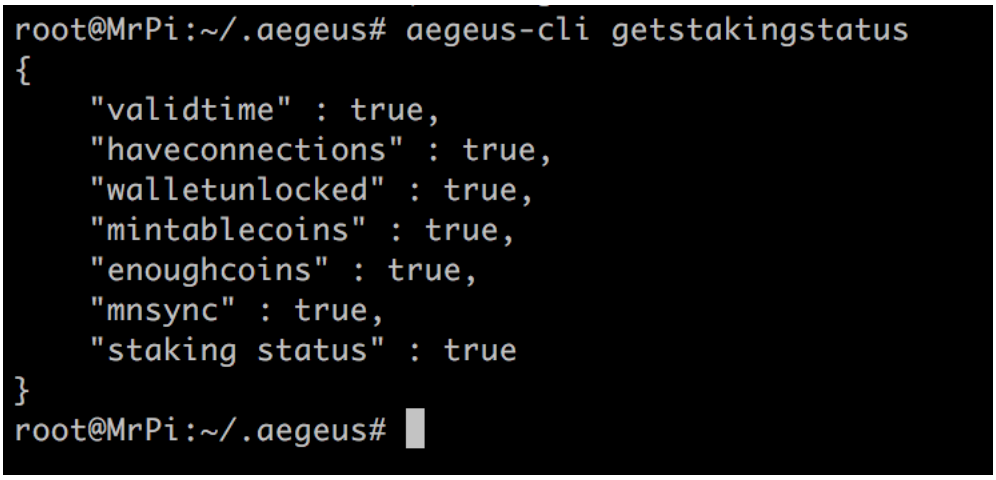
```
aegeus-cli getinfo
```

Unlock your wallet for staking only

```
aegeus-cli walletpassphrase <yourpassphrase> 0 true
```

Check if the wallet is staking

```
aegeus-cli getstakingstatus
```



All these outputs must show true for staking to be active

Note: If your coins were just transferred to a new address, it is possible that "mintablecoins" is false in the first 30 minutes or so.

Finally remove all command history from your Pi

```
history -c
```

6 Conclusions

Your Pi is now staking coins and will continue to do so. The easiest to follow the transactions is by using the blockchain explorer here: https://chainz.cryptoid.info/aeg/

Just input your coin address and see which transactions have been made to it

To see your addresses for your accounts on the Pi:

```
aegeus-cli listaddressgroupings
```

If you also want to also use your Pi as a controlling (cold) wallet for your masternode, please see Addendum B.

7 Addendum A - Add an existing address to your wallet

If you want to transfer an existing address from one wallet to another you can do that with the following steps:

Open your original wallet and open the debug console

Dump the private key for your address:

```
dumpprivkey <aegeusaddress>
```

Repeat this for every address you want to transfer.

If you don't know your addresses, use the following commands:

```
listaddressgroupings
```

NEVER give this private key to anyone and securely save it somewhere in case you need it to restore your wallet sometimes (like an offline USB stick in your safe deposit box)

If anyone gets hold of this key, they can steal your coins !!

To add these private keys to your Pi wallet, first unlock the wallet on your Pi (not only for staking, but fully)

```
aegeus-cli walletpassphrase <yourpassphrase> 0
```

Then add the private keys one by one

```
aegeus-cli importprivkey <aegeusprivkey>
```

Now check your account status with

```
aegeus-cli listaddressgroupings
```

If you don't see the balances yet, restart the Aegeus service

```
systemctl restart Aegeus.service
aegeus-cli walletpassphrase <yourpassphrase> 0 true
```

Check if the coins are staking correctly

```
aegeus-cli getstakingstatus
```

Everything must be ‘true’

Finally remove all command history from your Pi

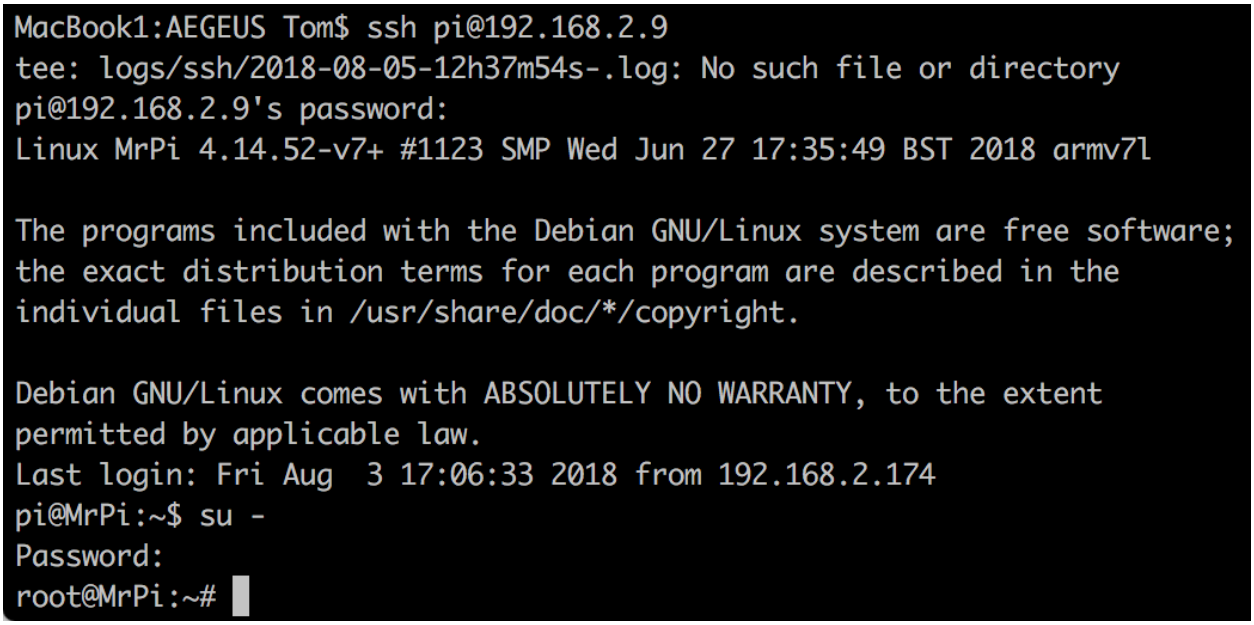
```
history -c
```

8 Addendum B - Control your masternodes

To configure your Pi as a control (cold) wallet to control your new or existing masternode, please follow the steps here

1. Open an SSH session to your Pi and become root (Use your favorite SSH tool for this like terminal, Putty, Bitvise, ...). The standard user for a Raspberry Pi running Raspbian is 'pi'

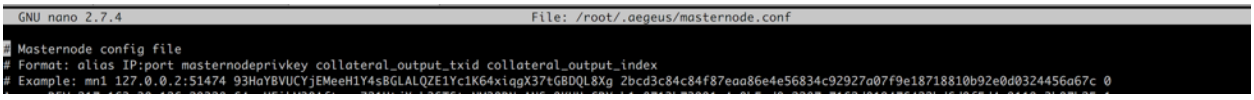
```
ssh pi@<IP address>
su -
```



2. Open the masternode configuration file

```
nano /root/.aegeus/masternode.conf
```

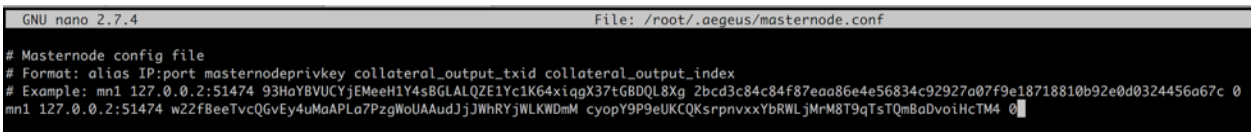
Start a new line for your config and enter it like shown in the example with one space between each part.



<SOME NAME> <IP ADDRESS> <MASTERNODE GEN KEY> <TX ID> <0 or 1>

So it would look something like:

mn1 127.0.0.2:51474
w22fBeeTvcQGvEy4uMaAPLa7PzgWoUAAudJjJWhRYjWLKWDmM
cyopY9P9eUKCQKsrpnvxxYbRWLjMrM8T9qTsTQmBaDvoiHcTM4
0



Exit nano with Ctrl+x - Yes - Enter

To repeat quickly, there are 5 pieces of information you use here:

The first one is an alias name of your choosing (can be anything).

The second one is that IP Address and Port number that come from the terminal after you ran the masternode installer script (you can find the guide for that here: https://aegeus.io/docs/MNGuideLinux.pdf)

The third part is the Masternode gen key that you got from the terminal after you ran the script.

The fourth part is the TX ID that you get from the cold wallet when you run the command

```
aegeus-cli masternode outputs
```

and finally the last part is either a 0 or 1 (whatever you got when you ran the “masternode outputs” command. Each of these must be separated with exactly one space.

Note: If masternode outputs is empty, then you don’t have an address with exactly 5000 coins in your wallet.

To finish adding your masternode, you need to restart your wallet (and unlock for staking again)

```
systemctl restart Aegeus.service
aegeus-cli walletpassphrase <yourpassphrase> 0 true
```

Now enable your masternode from the wallet

```
aegeus-cli masternode start-missing mn1
```

Check if the start was successful

```
aegeus-cli masternode list-conf
```

(Optional) If you want the masternode rewards to automatically combine to larger stacks for greater staking chances, you can use the following command:

```
aegeus-cli autocombinerewards true|false <threshold>
```

for example:

```
aegeus-cli autocombinerewards true 5
```

In this case, if there is a stack with more than 5 coins in it, it will automatically combine (to prevent dust <5 coins from trying to autocombine all the time)

(Optional)You can also chose to automatically split your stacks when they get too large to optimize your staking chances.

```
aegeus-cli setstakesplitthreshold <value>
```

For example use 2000 as value, then the stack will grow until it reaches 4000 and then it will split in two

Never forget to always check if your wallet is staking before you leave your SSH session:

```
aegeus-cli getstakingstatus
```

Tip: Use https://nodecheck.io to monitor your masternodes. It can send you a discord/telegram/mail when your masternode changes status or when a reward has been paid.