COLD WALLET + MASTERNODE SETUP ON LINUX + SENTINEL ON THE LINUX VPS

This tutorial shows the steps required to setup your IdaPay masternode on a Linux system while running a local cold wallet (Windows system here).

If you are lost and need help please consider helping me for my work by donating

Ida address: CqahpBzfmutg1sTKzEj5Lw8XUDhBvCWRpN

Let's get started!

CONTENTS

Pre-requirements

Generating your masternode private key

Preparing the server

Installing the daemon

Starting the daemon

Configuring the daemon

Transferring the funds to local (cold) wallet

Verifying the funds have been received

Configuring the masternode

Finalizing the wallet configuration

Starting your masternode

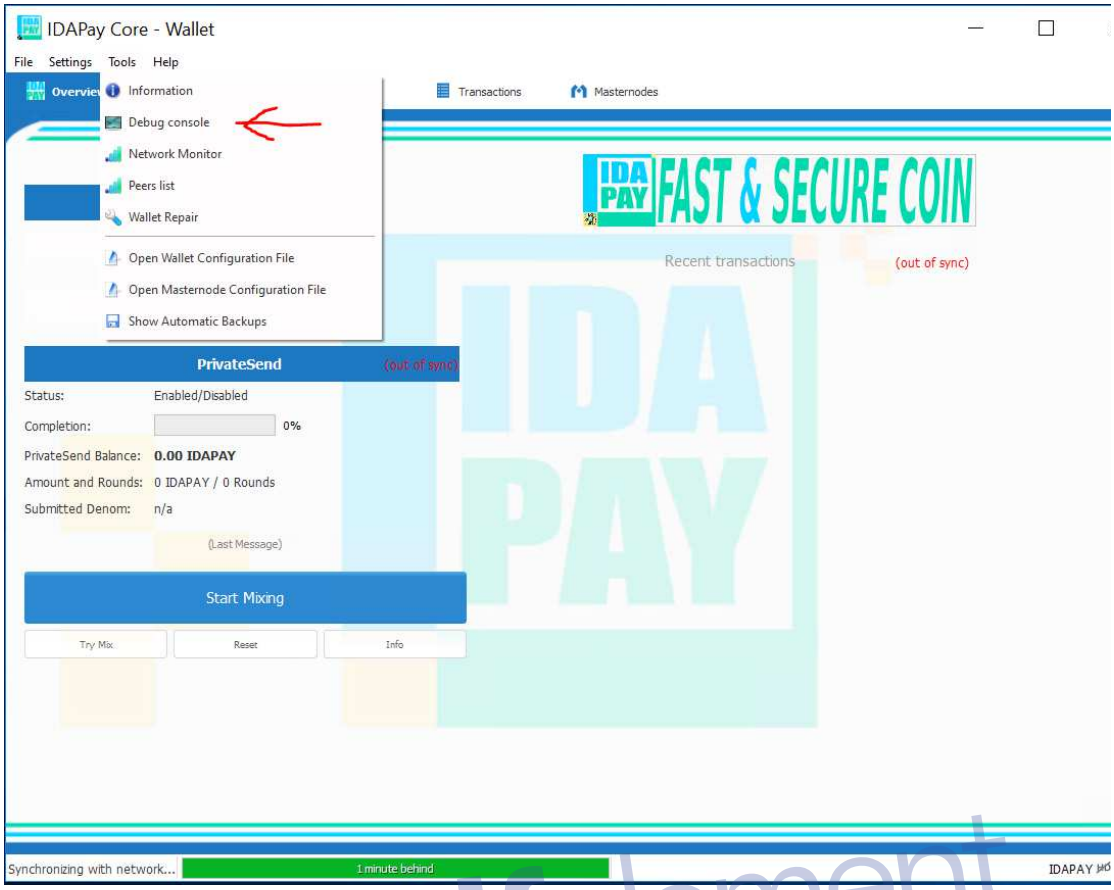
Linux server (e.g. Ubuntu 16.04 hosted on hetzner.cloud in this tutorial).

Windows Machine running idapay-qt.exe as local wallet.

Installing and setting up the sentinel on the linux VPS.

GENERATING YOUR MASTERNODE PRIVATE KEY

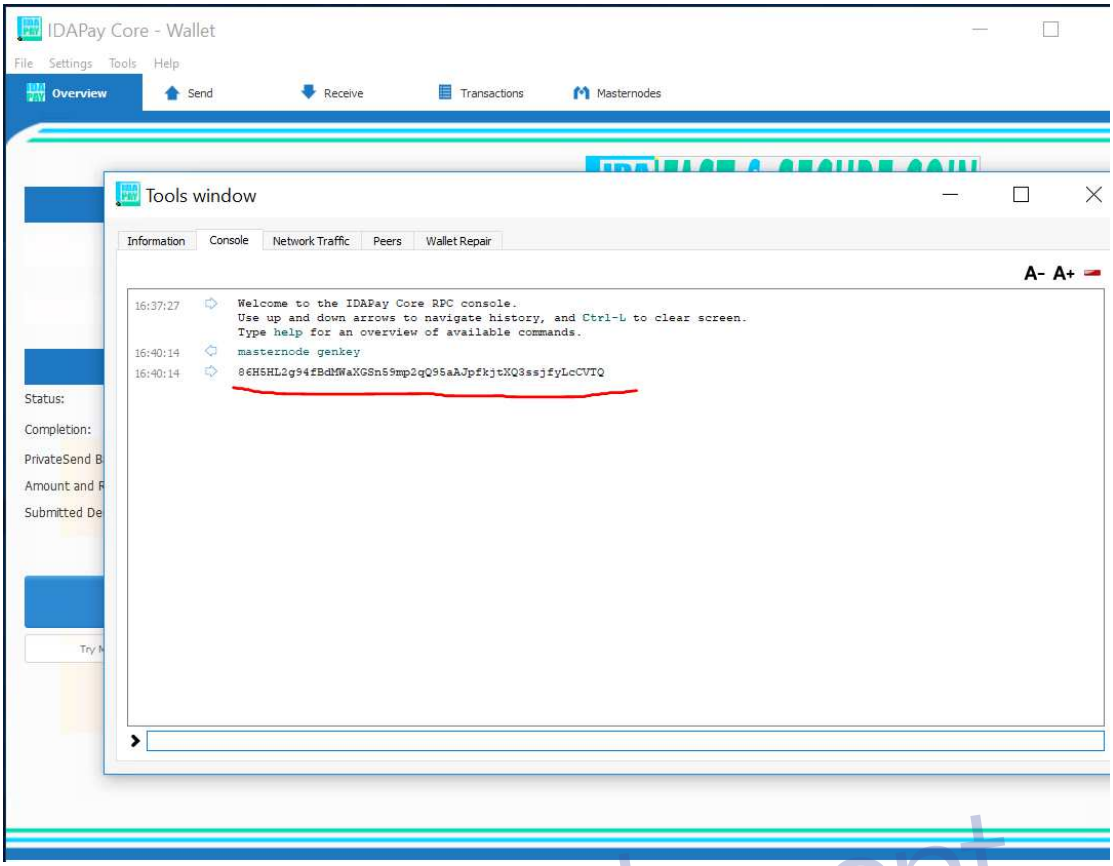
Open the Debug window from the Help > Debug Window menu.



Click on the Console tab and enter the following command:

"masternode genkey"

Please remember NEVER SHARE THIS KEY



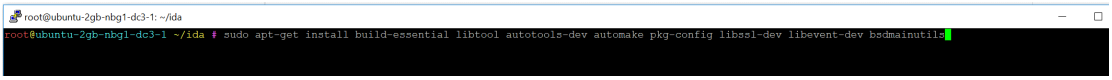
Keep the resulting MASTERNODE_PRIVATE_KEY as you will need it in following steps.

PREPARING THE SERVER

Open a shell and type the following commands to install the dependencies for the idapay daemon:

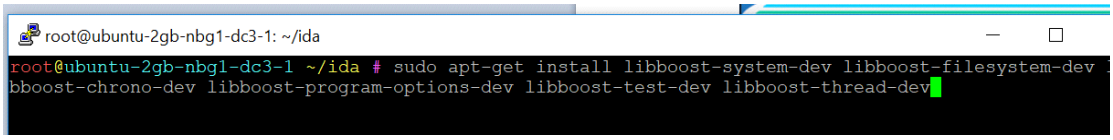
Build requirements:

sudo apt-get install build-essential libtool autotools-dev automake pkg-config libssl-dev libevent-dev bsdmainutils

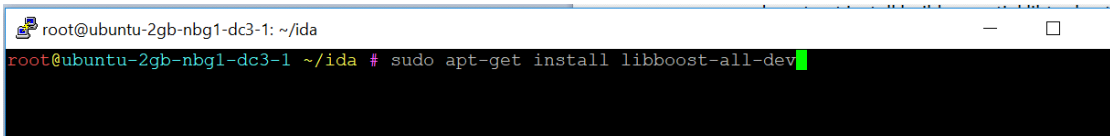


On at least Ubuntu 14.04+ and Debian 7+ there are generic names for the individual boost development packages, so the following can be used to only install necessary parts of boost:

sudo apt-get install libboost-system-dev libboost-filesystem-dev libboost-chrono-dev libboost-program-options-dev libboost-test-dev libboost-thread-dev



sudo apt-get install libboost-all-dev

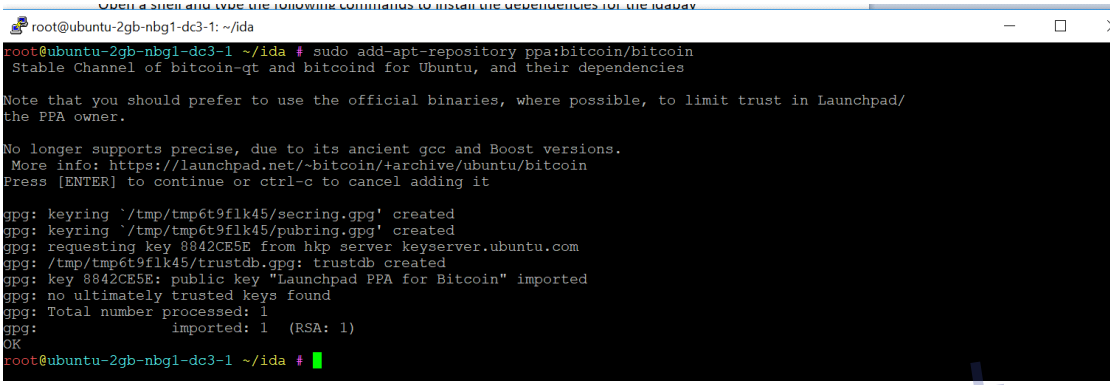


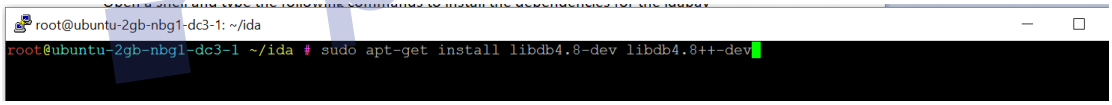
BerkeleyDB is required for the wallet. db4.8 packages are available here. You can add the repository and install using the following commands:

sudo add-apt-repository ppa:bitcoin/bitcoin

sudo apt-get update

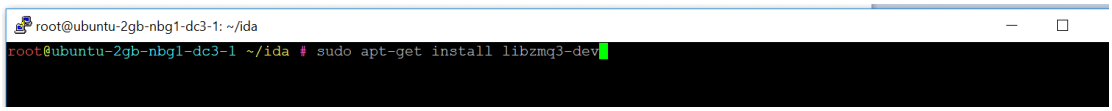
sudo apt-get install libdb4.8-dev libdb4.8++-dev





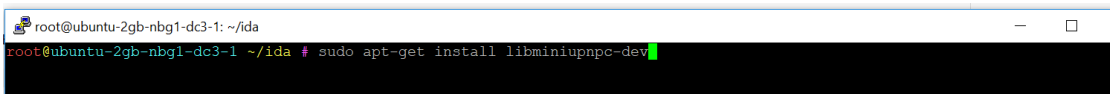
ZMQ dependencies:

sudo apt-get install libzmq3-dev



Optional:

sudo apt-get install libminiupnpc-dev

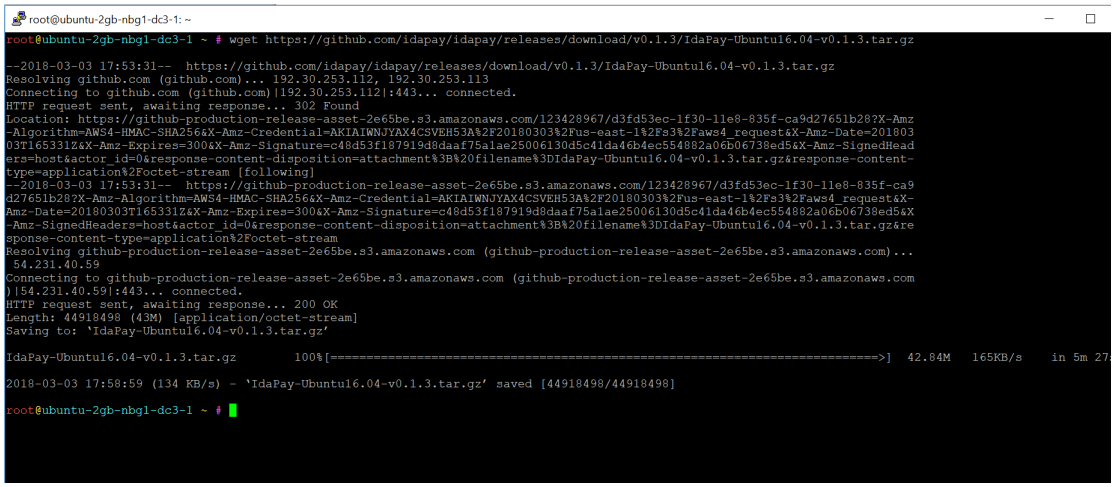


Next you need to download the latest version of the daemon wallet form the official idapay github page .

https://github.com/idapay/idapay/releases

In this case i am using IdaPay Release v0.1.3

wget https://github.com/idapay/idapay/releases/download/v0.1.3/IdaPay-Ubuntu16.04-v0.1.3.tar.gz



Next step you need to extract ne tar.gz archive with the next command:

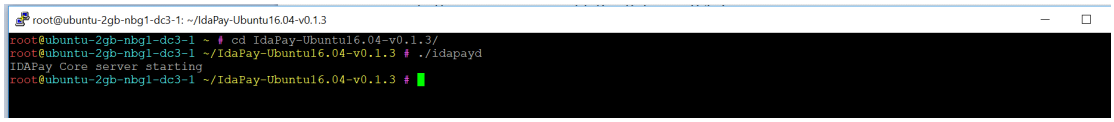
tar -xvzf IdaPay-Ubuntu16.04-v0.1.3.tar.gz

STARTING THE DAEMON

We initially run the daemon so it creates the initial configuration files:

cd IdaPay-Ubuntu16.04-v0.1.3/

./idapayd

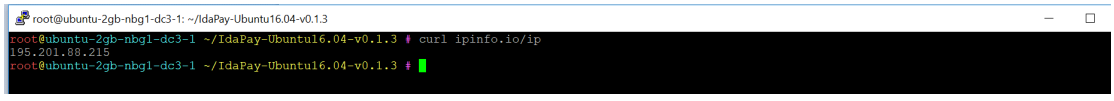


CONFIGURING THE DAEMON

You will need the following information in the next step:

EXTERNAL_IP: Your masternode public IP (if you don't know it simply

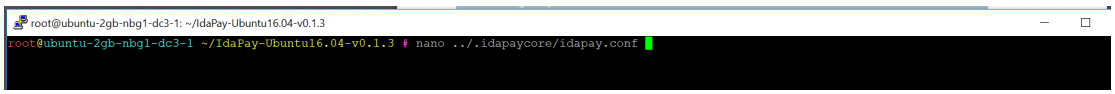
type the command curl ipinfo.io/ip in the shell to get it).



MASTERNODE_PRIVATE_KEY: The key generated from Windows wallet in first step.

Type the following command to edit the configuration file (using nano editor).

nano ../.idapaycore/idapay.conf



Edit the config file with all the information we collected in the previews steps

port=19285 (must be this port)

masternode=1

masternodeprivkey=85zhkHsabVuKNXyn6AK7N4TAu4Uss4soYVuEqEMDoCLakMihvg3 (we collected that key from your windows wallet in previews steps)

daemon=1

server=1

externalip=195.201.88.215:19285 (we got the VPS external ip with the curl ipinfo.io/ip command in the previews steps)

listen=1

rpcuser=anyuseryouwant

rpcpassword=anypasswordyouwant

rpcallowip=127.0.0.1

rpcallowip=81.157.7.157 (this is the ip on your windows computer to find it if you dont know it just google "what is my ip")

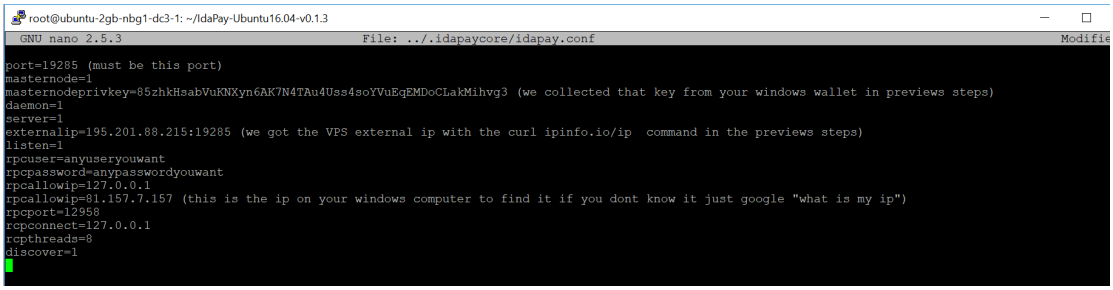
rpcport=12958

rcpconnect=127.0.0.1

rcpthreads=8

discover=1

So your config looks like that:



Now you can exit the editor with Ctrl+x followed by Y to confirm

We can now start the daemon again and it should print the message below:

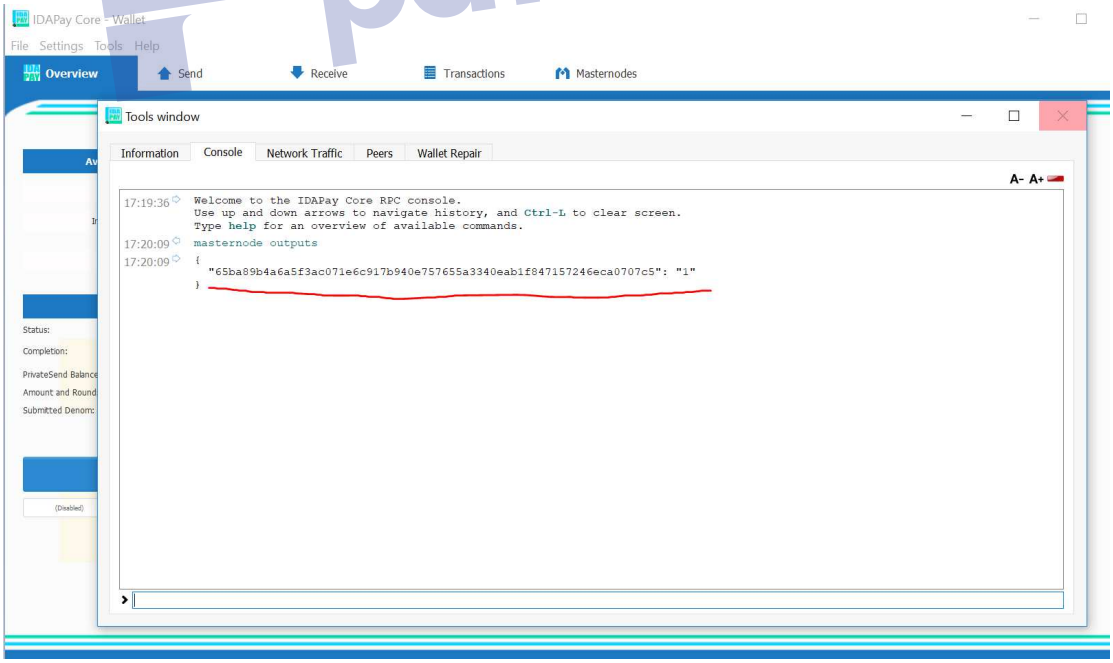
./idapayd

TRANSFERRING THE FUNDS TO LOCAL (COLD) WALLET

Back to the local wallet on Windows, you can now transfer the funds to it. You can use the default address or use the Receive Tab to create a new address. Send exactly 1500 IDA to this address.

Keep track of the transaction id. You can retrieve the transaction id if you go to Tools-> Debug console and type:

"masternode outputs"



CONFIGURING THE MASTERNODE

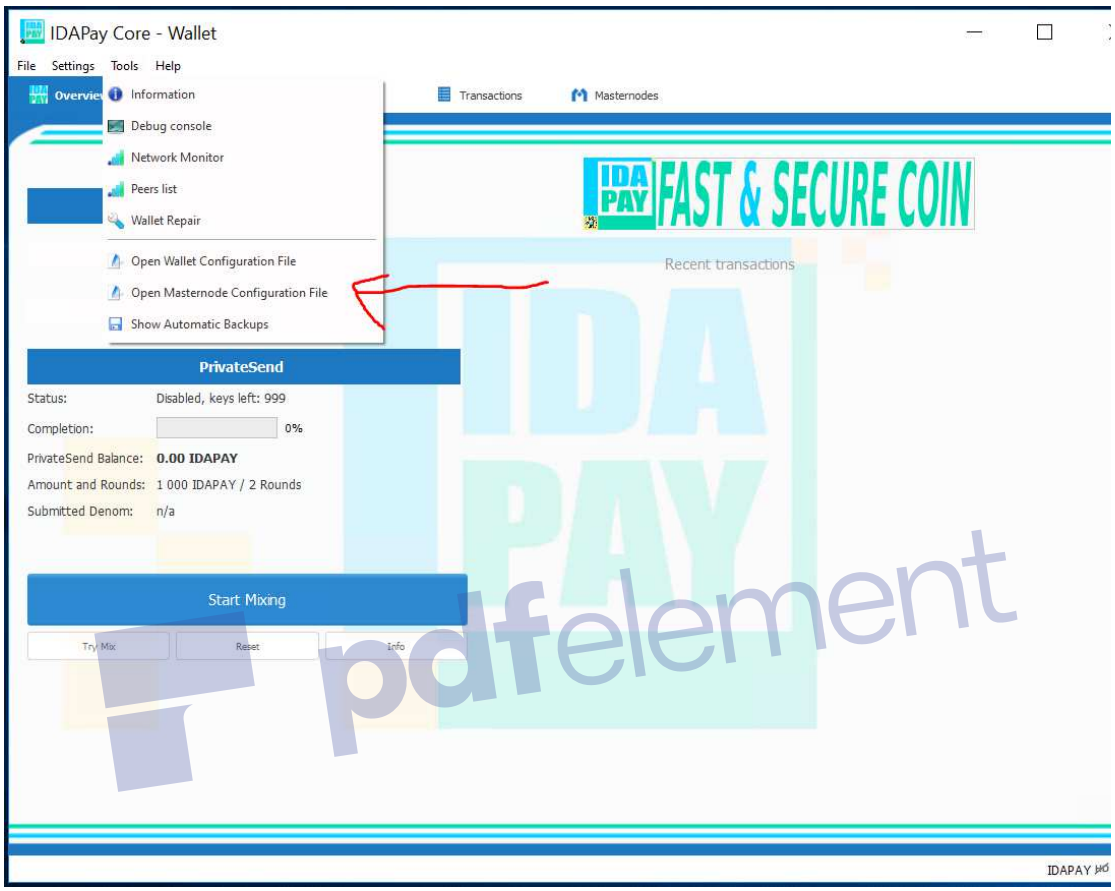
You will need the following information to complete the next step:

MASTERNODE_PRIVATE_KEY: The key generated from Windows wallet in first step.

TRANSACTION_ID: The transaction ID from the funds generated in previous step.

OUTPUT: The number (1 or 0) printed after the Transaction ID in the previous step

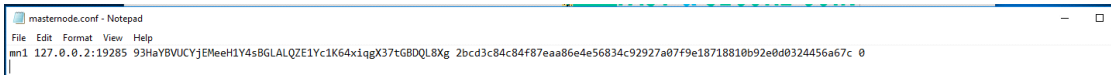
Next step will be to open the masternode config file by clocking tools -> open masternode configuration file :



Copy the line below (Replace with information gathered previously):

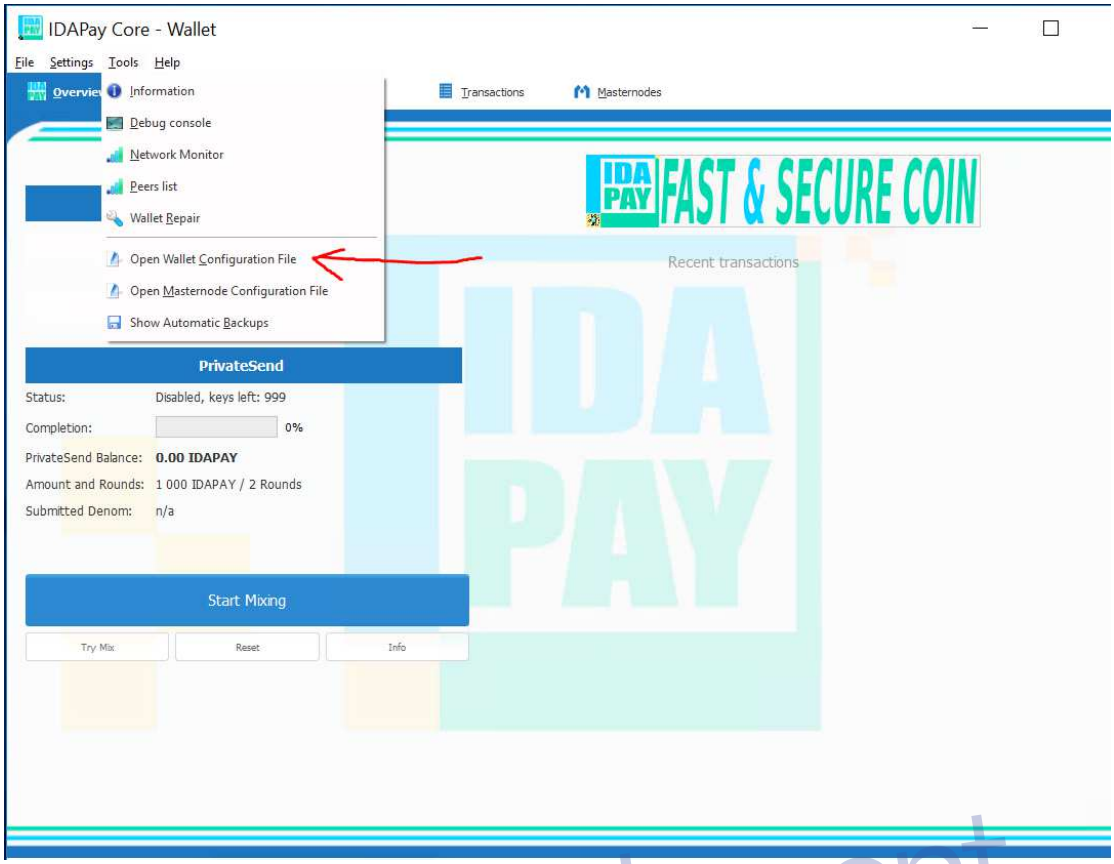
mn1 EXTERNAL_IP:19285 MASTERNODE_PRIVATE_KEY TRANSACTION_ID OUTPUT

It should look like this:



FINALIZING THE WALLET CONFIGURATION

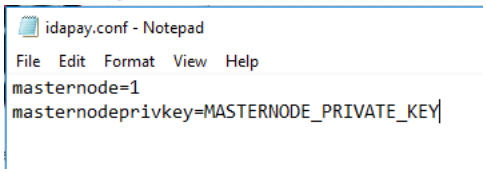
menu Settings > Open wallet configuration and add the following entries:



masternode=1

masternodeprivkey=MASTERNODE_PRIVATE_KEY

The file should look like this:

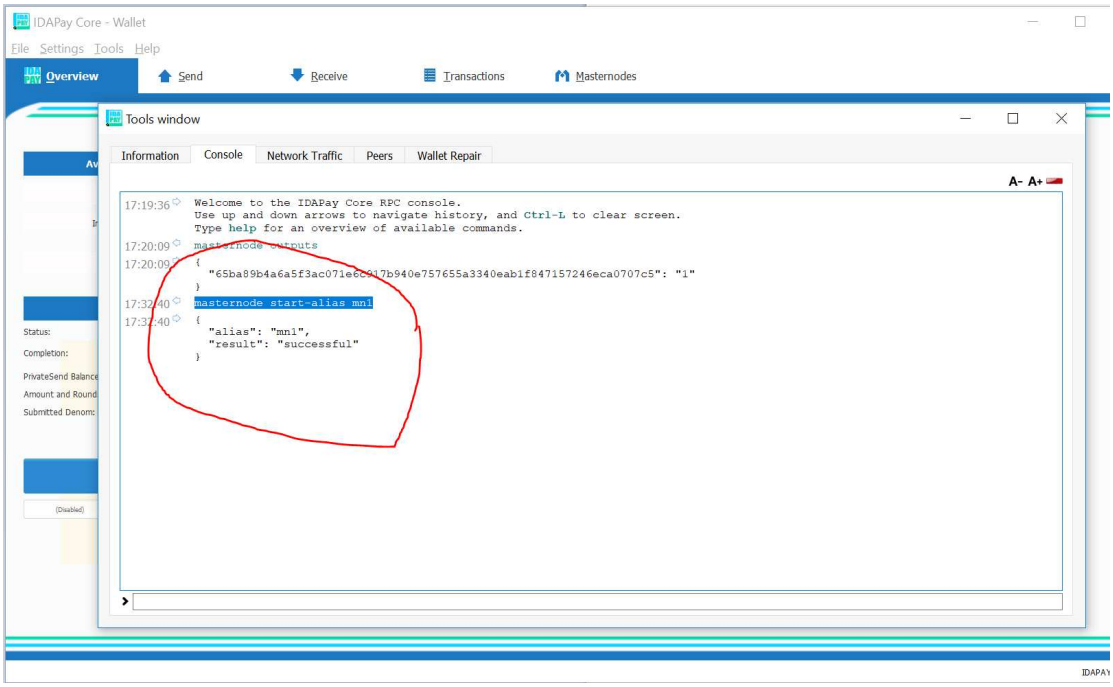


STARTING YOUR MASTERNODE

Restart idapay-qt.exe to pick up the configuration changes.

Using the Debug Window on local wallet, type the following command:

masternode start-alias mn1



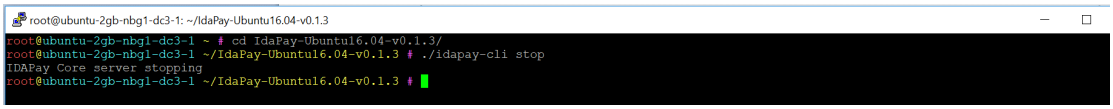
At that point your masternode will show watchdog expired . Please do not panic this is normal all it means is we need to go back to the VPS and install the sentinel watchdog.

So back on the VPS.

Before starting the steps make sure to restart your masternode with reindex so you are using the fresh blockchain. Delete mncache.dat and mnpayments.dat from your idapaycore folder before reindexing.

cd IdaPay-Ubuntu16.04-v0.1.3/

./idapay-cli stop

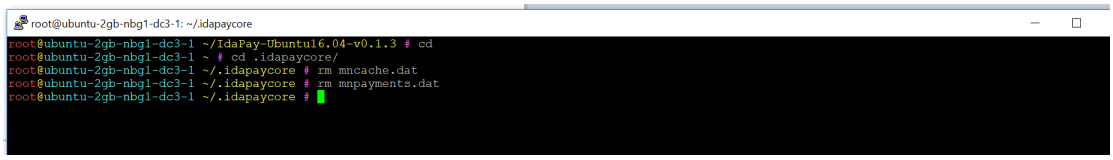


cd

cd .idapaycore/

rm mncache.dat

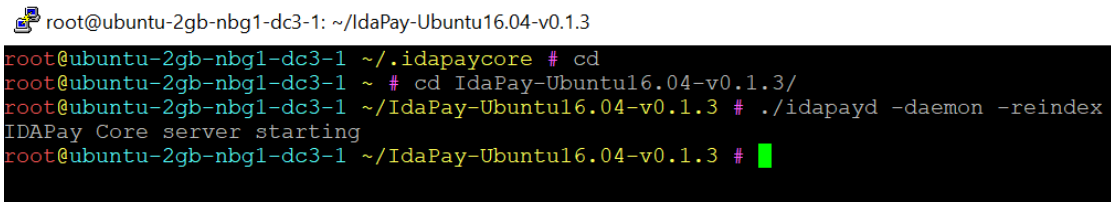
rm mnpayments.dat



cd

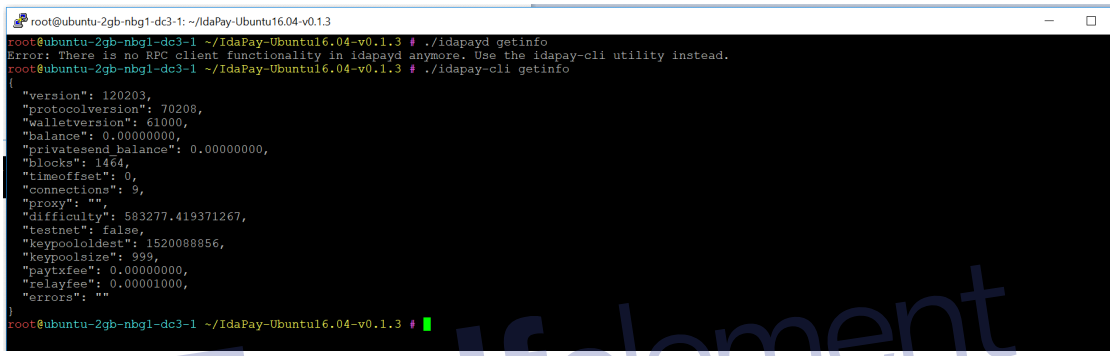
cd IdaPay-Ubuntu16.04-v0.1.3/

./idapayd -daemon -reindex



Wait for the full resync to the latest block. You can check regularly with :

./idapay-cli getinfo

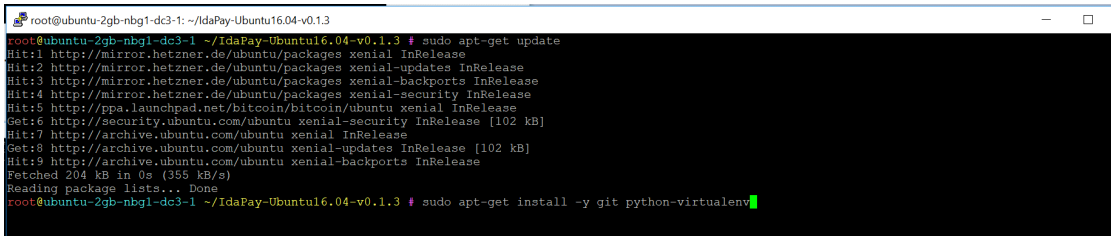


Once your wallet is fully synced you are now ready to follow the steps below:

Install the prerequisites:

sudo apt-get update

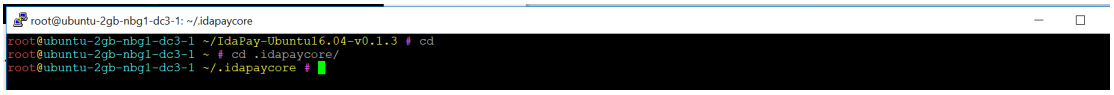
sudo apt-get install -y git python-virtualenv



If you are not already there, navigate to your .idapaycore folder:

cd

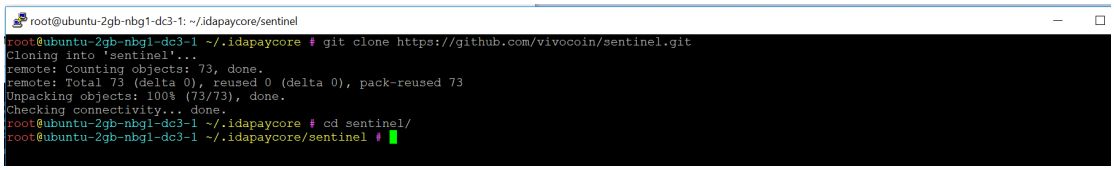
cd .idapaycore/



Clone sentinel, switch to the sentinel directory

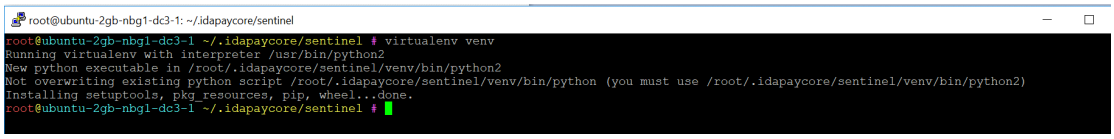
git clone https://github.com/vivocoin/sentinel.git

cd sentinel



Create virtual python environment

virtualenv venv

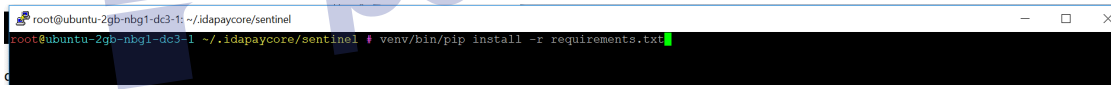


If this command fails try installing this package :) :

sudo apt-get install -y virtualenv

Install sentinel dependencies

venv/bin/pip install -r requirements.txt



Test sentinel is alive and talking to the still syncing wallet

Open sentinel.conf:

"nano sentinel.conf"

remove the comment and edit the line to point sentinel to your idapay config file

vivo_conf=/root/.idapaycore/idapay.conf

the conf should look like this:

Now you can exit the editor with Ctrl+x and confirm with Y.

Next step is to run the sentinel watchdog:

venv/bin/python bin/sentinel.py

You should see: “idapayd not synced with network! Awaiting full sync before running Sentinel.”

This is exactly what we want to see at this stage

Wait until the reindex has complete and the wallet has sync’d

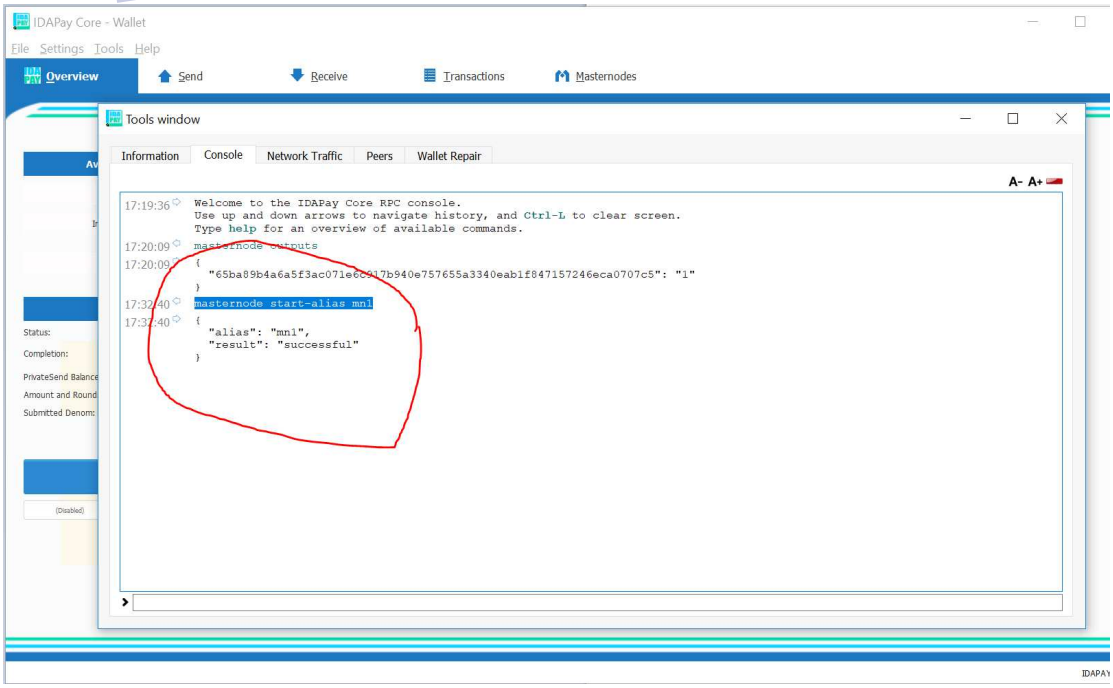
At this point, your remote masternode is synchronized and chatting with the network

but is not accepted as a masternode because it hasn’t been introduced to the network

by your collateral.

You may now start your masternode! Go back to your local wallet, open the debug console, and run these commands (LABEL is the name you used for your MN in the masternode.conf):

"masternode start-alias mn1"



This will start your masternode (hopefully :))

Test sentinel has nothing bad to say:

You're needed back in Sentinel directory! :D

cd

cd .idapaycore/sentinel/

At this point, running:

venv/bin/python bin/sentinel.py

should return nothing but silence. This is how you know it's working, and your masternode is working.

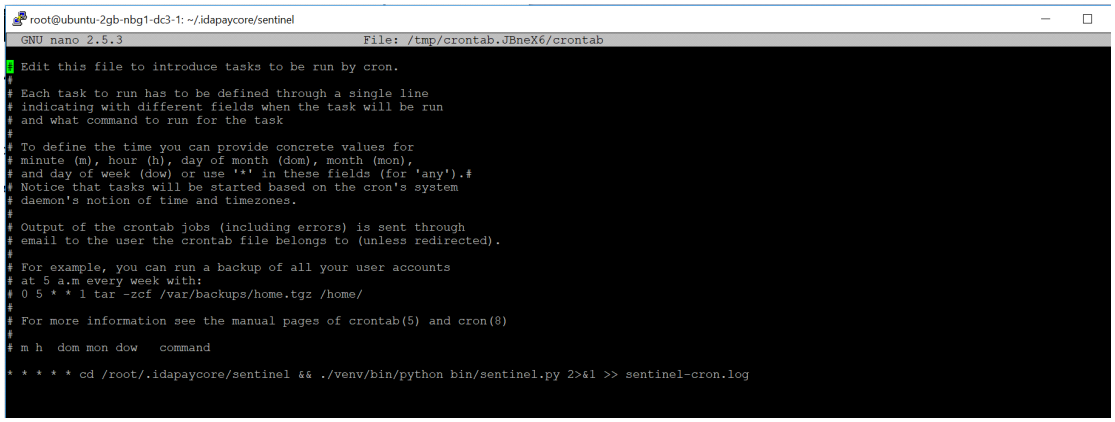
Create a crontab entry to wake sentinel every five minutes

crontab -e

Choose nano as your editor(option 2). Add this line to the end of the file.

* * * * * cd /root/.idapaycore/sentinel && ./venv/bin/python bin/sentinel.py 2>&1 >> sentinel-cron.log

Your crontab should look like that.



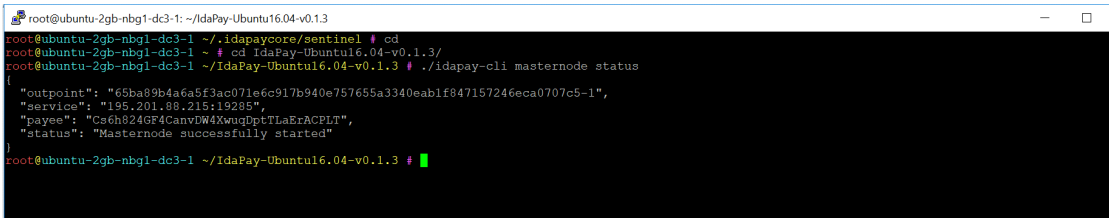
Now exit the editor with Ctrl+x and confirm with Y

Lets check if everything is working

cd

cd IdaPay-Ubuntu16.04-v0.1.3/

./idapay-cli masternode status



And that's it, you've made it! You can feel like a Crypto-GOD, rocking your new Masternode!

This guide was created by Valentin Danev (pool.ddclub.org)

If you found my work of any help at all please consider donating that will help me to keep working and helping the comunity my donations ida address is :

CqahpBzfmutg1sTKzEj5Lw8XUDhBvCWRpN