Masternode Setup Guide

Preface
The following guide to set up a Bitcreds Masternode is geared towards beginners with no experience of servers nor of Bitcreds.
You can skip certain parts if you're already familiar with them.

Requirements
In order to set up a Masternode you will need:
- A local Bitcreds wallet installed on your main machine containing 10,000 or more Bitcreds to cover the Masternode collateral cost and any possible fees for transactions.
- A VPS with a fixed IPv4 address with the port 31000 opened.
(If the VPS has both IPv4 and IPv6, this line: "*bind=<IPv4 address>*" should be added to the bitcreds.conf file.)

What are the hosting requirements for hosting a Masternode?
The hosting requirements for hosting a Masternode are very low as the wallet itself needs very few system resources.
A VPS with 1 vCore, 1024 MB RAM and a 25 GB SSD should be more than sufficient.

Which OS should I use?
You can use any 64-bit OS based on Linux you want, but I would recommend using Ubuntu 16.04 because of two reasons:
- It's quite beginner friendly.
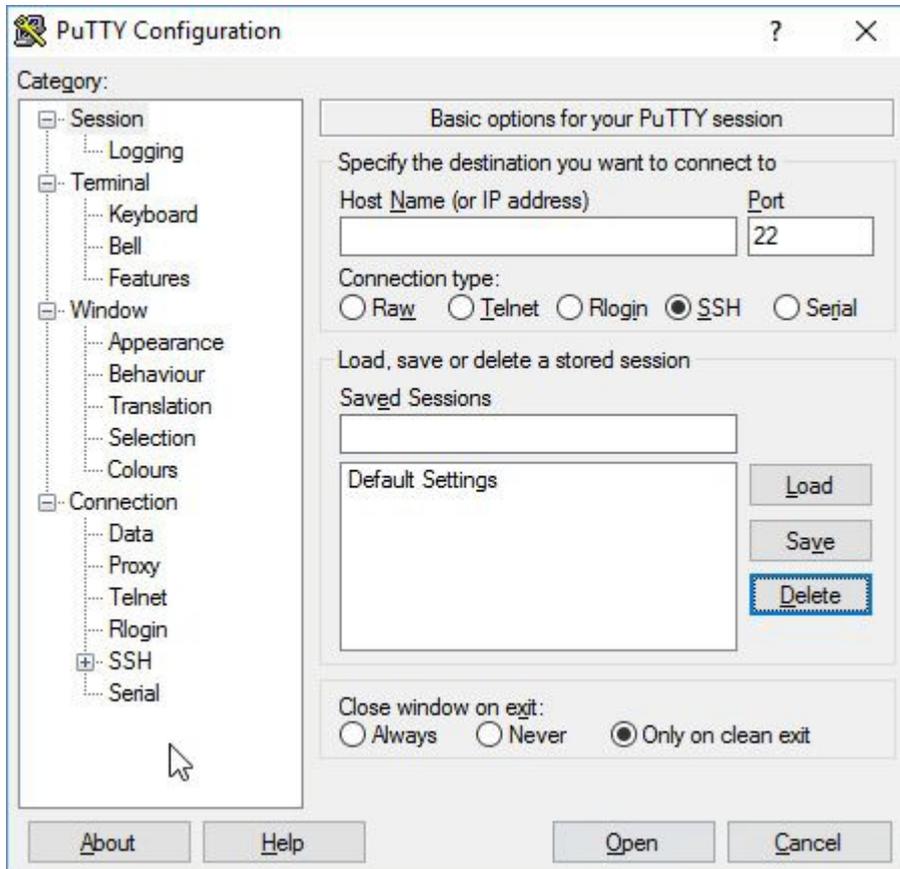- This guide is based on Ubuntu 16.04.

Step 1.1: Download and install PuTTY
You can download PuTTY at http://www.putty.org/. The installation should be pretty self-explanatory.

Step 1.2: Connect to your VPS
You should have been provided with the login credentials for your VPS from your hosting provider, those are needed now.

Start PuTTY now, the window should look like this:

Type the IP address of your VPS in the field below "Host Name (or IP address)" and choose SSH as the "Connection type", afterwards click on open. A new window (called Terminal) should open:
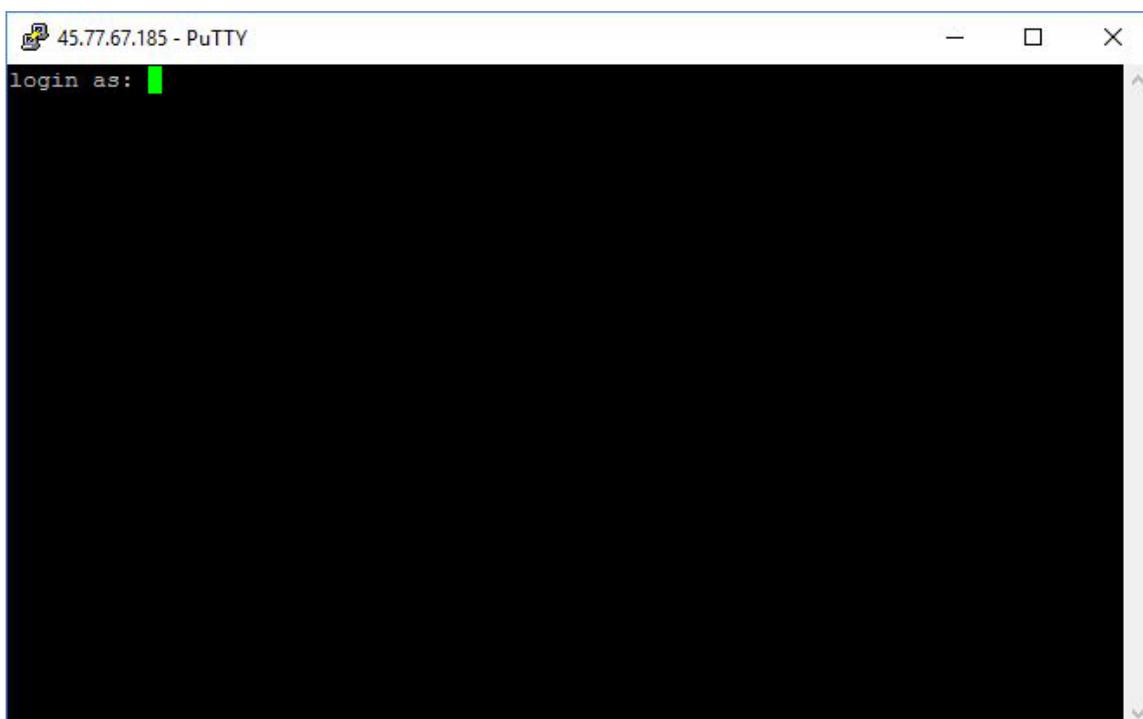
Your VPS credentials are now needed again:

1. Type your username.
2. Press enter.
3. Copy your password and right-click in the PuTTY window (don't worry if you can't see anything written, it's normal).
4. Press enter.

You are now logged in to your VPS.

Step 1.3: Make sure that the time is correct on your server

Type the following commands into your Terminal (Ubuntu/Debian):

sudo apt-get install ntp

sudo systemctl enable ntp

On CentOS servers you can do it this way: *yum -y*
install ntp ntpdate ntp-doc systemctl enable
ntpd.service ntpdate pool.ntp.org
service ntpd start

Step 1.4: Get the Bitcreds Linux wallet

***Optional** – You can save the time it takes to sync by installing a bootstrap of the blocks first by following the instructions in*
https://bitcreds.org/documents/Bitcreds%20Bootstrap%20Instructions.pdf

To download the Bitcreds wallet for Linux, type the following command into PuTTY:

wget https://github.com/Bitcreds/Bitcreds/releases/download/v1.7.1.0/bitcreds-v1.7.1-linux.tar.gz

After that, we need to unpack the archive by typing this command into the Terminal:

tar -xzf bitcreds-1.7.1-linux.tar.gz

Step 1.5: Enabling the firewall and opening port 31000

For the Masternode to work, we need to have port 31000 opened. Before we can open port 31000, we should enable the firewall (to make the server a bit more secure).

You can enable the firewall by typing this command into the Terminal:

sudo ufw enable

This warning will come up: "*Command may disrupt existing ssh connections. Proceed with operation (y/n)?*". Type "y" and press enter.

To ensure that we can reconnect to our VPS, we need to open port 22 (the SSH port) first.

You can do this by running this command:

> *sudo ufw allow 22*

Now, we can open the required port for the Masternode by running this command:
> *sudo ufw allow 31000*

Step 1.6: Starting the Bitcreds daemon (basically the wallet)
We need to navigate to the Bitcreds directory first by running this command:
> *cd bitcreds-v1.7.1-linux*
By running this command, you can also see the binaries:
> *ls -lisa*

To start the Bitcreds daemon, run the following command:

> *./bitcredsd --daemon*

You can now check the status of the Bitcreds daemon by typing:

> *./bitcreds-cli getinfo*

It should look something like this:

```
{
  "version": 1070100,
  "protocolversion": 70010,
  "walletversion": 60000,
  "balance":  0.00000000,
  "privatesend_balance": 0.00000000,
  "blocks": 1002621,
  "timeoffset": 0,
  "connections": 49,
  "proxy": "",
  "difficulty": 0.0002338534463307286,
  "testnet": false,
  "keypoololdest": 1613280764,
  "keypoolsize": 1999,
  "paytxfee": 0.00000000,
  "relayfee": 0.00010000,
  "errors": ""
}
```

The Bitcreds daemon should automatically start syncing the blockchain (though it can take a while for it to start syncing).

This can be seen here by:

> *"blocks": 2304*

You will now have to wait until the wallet has finished syncing. It will be finished when you run:

> *./bitcreds-cli getinfo*

Repeatedly a few times and the number of blocks don't change anymore.

Step 1.7: Generate a private key for the Masternode
Next, we need to create a private key for the Masternode. The controller wallet on your local machine will use that private key to start/communicate with your Masternode.

You can generate the private key by running this command:

> *./bitcreds-cli masternode genkey*

Copy this key (you can do this in PuTTY by clicking the left mouse button in the Terminal window and dragging to select the text. When you let go of the mouse

button, the text is automatically copied to the clipboard) and paste it somewhere (e.g. a text document).

You can now stop the Bitcreds daemon by running this command:

> *./bitcreds-cli stop*

Step 1.8: Edit the Bitcreds configuration file
Open up the Bitcreds configuration file by typing:

> *nano ~/.bitcreds/bitcreds.conf*

Your terminal should look like this after you have run the command:

Add the following lines to the Bitcreds configuration file:

> *onlynet=IPV4*
> *externalip=<your VPS IP>*
> *masternode=1*
> *masternodeprivkey=<the key you generated in step 1.7>*

(If you have copied the key/have it in your clipboard, you can just right-click in the Terminal to paste it).

Now, save the Bitcreds configuration file with CTRL + O + ENTER and close the file with CTRL + X.

Step 1.9: Starting the daemon

Run this command to start the daemon:

> *./bitcredsd --daemon*

You can check the status of your Masternode by running this command:

> *./bitcreds-cli masternode status*

You should see something like this:

```
{
  "vin": "CTxIn(COutPoint(0000000000000000000000000000000000000000000000000000000000000000, 4294967295), coinbase )",
  "service": "[::]:0",
  "status": "Node just started, not yet activated"
}
```

The wallet on your local machine acts as the controller wallet for your Masternodes. It contains the Masternode collateral and can activate your remote Masternode on your VPS.

Step 2.1: Setting the collateral funds

In the Bitcreds wallet, we will now need to go to the "Receive" tab to create a new address.
You may use a label like "Masternode Collateral". Set the
"Amount:" to exactly 10000.
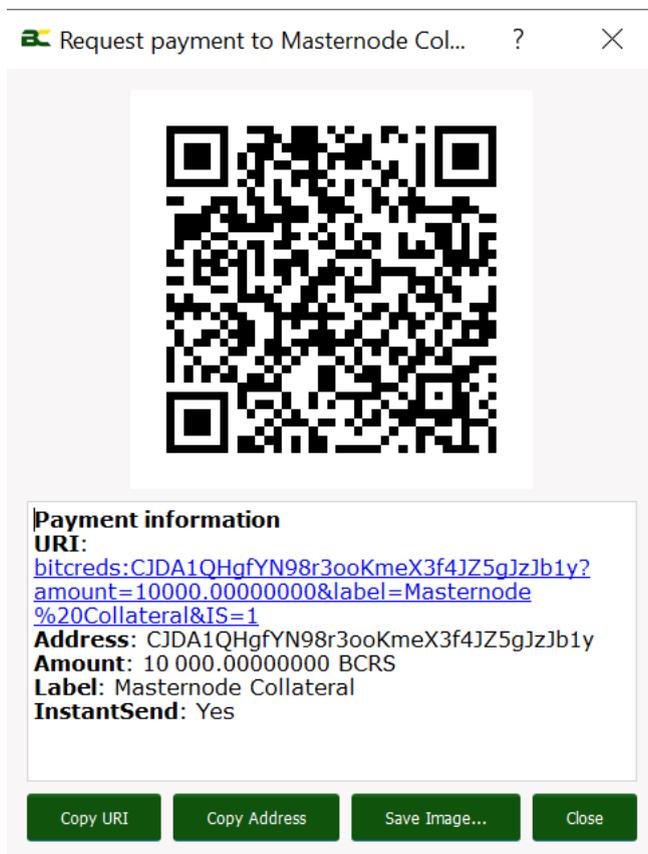A message is not needed.
Let the checkboxes below the text fields untouched and click on "Request payment".
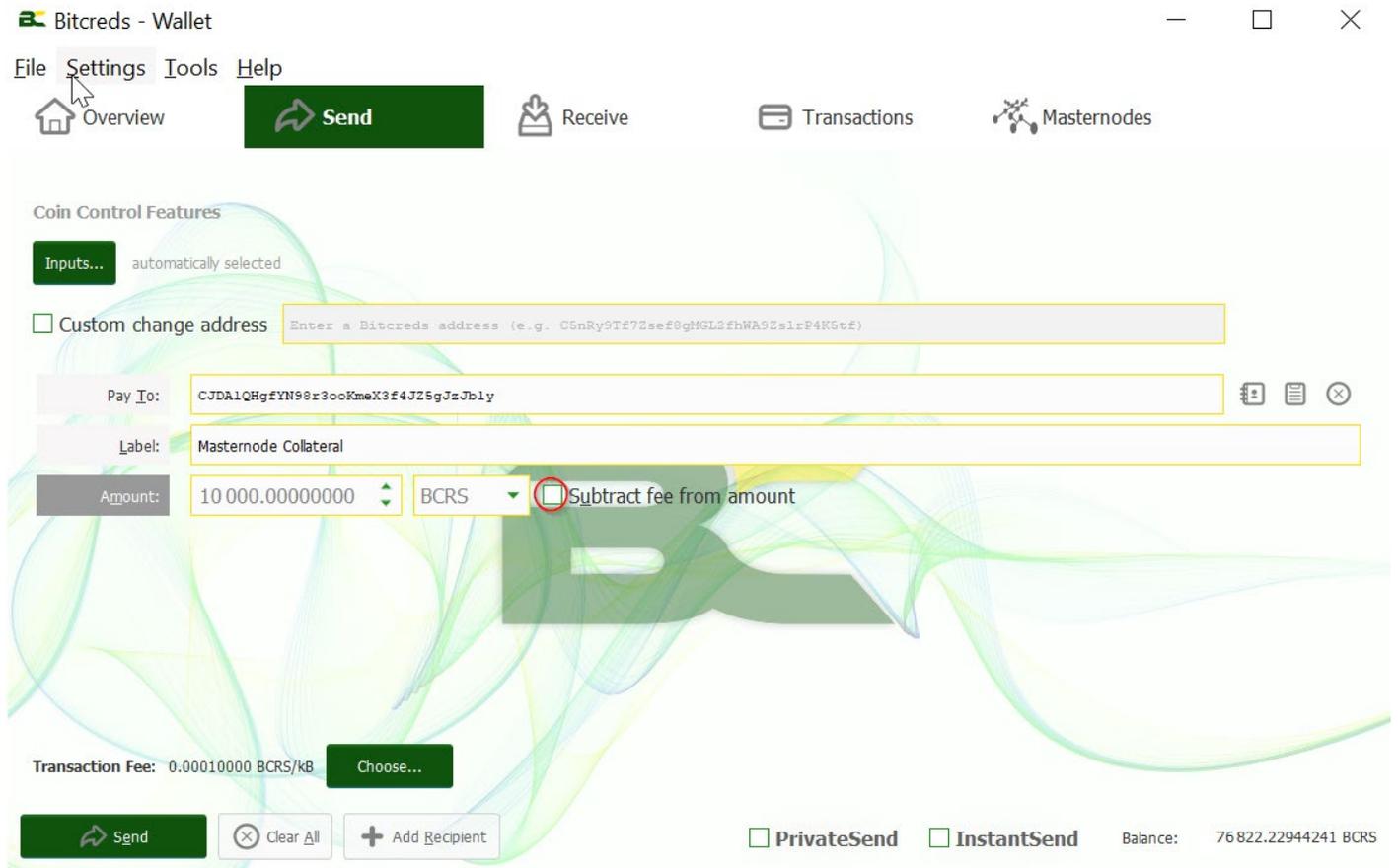
A new window like this one will appear:

Copy the address in your window and go to the "Send" tab.

Paste your address into the "Pay To": text field and set the "Amount:" text field to exactly 10000 Bitcreds.

Make sure the "Subtract fee from amount" checkbox is NOT checked.

Then click on the "Send" button in the bottom left corner.

In the "Confirm send coins" window click "Yes".

Now, go to "Tools" => "Debug Console" and run this command:

masternode outputs

You should see something like: "<collateralTXID>":"<TXINDX>".
The <collateralTXID> is a long alphanumerical string, whereas the <TXINDX> is a small number, usually 1 or 0.
Copy and paste these somewhere.

Step 2.1.1: Confirming the <collateralTXID>
To confirm that this particular <collateralTXID> belongs to your Masternode payment you need to copy and paste it into the Bitcreds block explorer. (http://explorer.crds.co/).
If you see a payment to an address of exactly 10,000 Bitcreds then that's most likely a Masternode collateral payment. To confirm, copy the address that received the 10,000 Bitcreds and paste it into the "Pay To" field on the "Send" tab.

If the "Label" field changes its text to the label specified in step 2.1 (here: "Masternode Collateral"), you can be sure that this <collateralTXID> correctly belongs to your Masternode collateral payment.

Step 2.2: Configuring the Masternode configuration file on the controller wallet
Go to "Tools" => "Open Masternode Configuration File" and add this line:

<alias> <VPS IP>:31000 <masternodeprivkey> <collateralTXID> <TXINDX>

- <alias>: Any name can be given (make sure your <alias> doesn't contain spaces, rather use "_")
- <VPS IP>: The external IP address of your VPS.
- <masternodeprivkey>: The key you generated in step 1.7.
- <collateralTXID>: The long alphanumerical string you generated in step 2.1.
- <TXINDX>: The small number you generated in step 2.1.

An example is also available in the Masternode Configuration File.

Step 2.3: Enabling your Masternode
After you have saved and closed the Masternode Configuration File, restart your Bitcreds wallet.

Go to the "Masternodes" tab. Your Masternode will be listed with its alias as "MISSING".

You will now have to wait for your transaction to have at least 16 confirmations. You can see the status of the transaction by going to the "Transactions" tab, right-clicking on the first transaction on the top and then clicking on "Show transaction details".

Immediately after you have sent the transaction the status will be

   "Status: 0/unconfirmed"

After a little while, the status should change to:

   "Status: 16 confirmations"

Now, with 16 confirmations, we are ready to start our Masternode.
To do this, go to the "Masternodes" tab and right-click on your Masternode and click on "Start Alias".

The status of your Masternode should now change from "DISABLED" to "PRE_ENABLED",

after which it takes about an hour before the status changes to "ENABLED".

3.1: Note about setting up multiple Masternodes

Let's say you set up *server1* as your first Masternode and it has an IP address of *10.1.1.1*.

When you send the 10,000 Bitcreds to the receive address you created for *server1* and run *masternode outputs* in the Debug Console, you'll get the <collateralTXID> for *server1*.

Now, when you edit your masternode.conf file, you'll put in:

> *<alias> <VPS IP>:31000 <masternodeprivkey> <collateralTXID> <TXINDX>*

When you save the masternode.conf file and re-open the wallet, the blockchain ties 10.1.1.1:31000 (*server1*) to the specific <collateralTXID> you first used.

So, when you add a second Masternode *server2* with an IP address of 10.1.1.2, send the transaction of 10,000 Bitcreds to the receive address and run *masternode outputs* in the Debug Console, you'll have two transactions listed.

The outputs don't specify which transaction is for which server. However, if you use the first <collateralTXID> for *server2* and the second <collateralTXID> for *server1*, the wallet will show both Masternodes as being on the same IP address.

The important thing now is that you make sure that you only add a single Masternode at a time and that you are very specific with your own external record keeping (keep track of which <collateralTXID> belongs to which Masternode using an Excel Spreadsheet for example) to ensure that only the newest <collateralTXID> is tied to the newest Masternode you're setting up.