CROWDCOIN VPS setup guide

This is a step-by-step guide to installing a masternode daemon on a Linux VPS. This supposes a cold wallet setup, meaning a wallet application installed on another computer, typically Windows. This guide does not cover the installation of windows wallet, nor the setting up of its *masternode.conf* file.

The wallet application can be [found here](#).

In this guide, all commands that must be typed on VPS shall be displayed as such:

`this is a command`

Type this command exactly (you may copy&paste it). There will always be some space between commands, so that you can easily see commands spanning over several lines. Some commands may also be appended together with `&&`. You may execute these commands as one, or you may do each one separately.

One final note: this guide supposes you have at least a basic knowledge of linux commands.

Let's get started!

Set up an Ubuntu 16.04 instance from any provider

The cheapest one will do, provided you create a swap file (see below). Here are a few providers:

[Vultr](#)
[Hetzner](#)
[OVH](#)
[ArubaCloud](#)

Once you have access to your VPS, log in as root and prepare your VPS for handling a masternode. This requires updating the system, installing some packages and libraries as well as configuring the firewall.

Updates, libraries and packages to install

Update Linux installation (can take a few minutes to complete):

`sudo apt-get update && sudo apt-get upgrade -y && sudo apt-get dist-upgrade -y`

Reboot your VPS for changes to take effect:

`sudo reboot`

Install the following packages and libraries (might not necessary if you don't compile sources but good idea to do it so you got them installed anyways):

`sudo apt-get install -y ufw wget git nano htop unzip autoconf automake fail2ban`

`sudo apt-get install -y libboost-dev libboost-system-dev libboost-filesystem-dev libboost-chrono-dev libboost-program-options-dev libboost-test-dev libboost-thread-dev libboost-all-dev libminiupnpc-dev`

```
sudo apt-get install -y build-essential libtool autotools-dev pkg-config
libssl-dev libevent-dev bsdmainutils virtualenv python-virtualenv
```

```
sudo apt-get install -y software-properties-common && add-apt-repository
ppa:bitcoin/bitcoin
```

```
sudo apt-get update -y
```

```
sudo apt-get install -y libdb4.8-dev libdb4.8++-dev libzmq3-dev libzmq5
```

Set up SWAP FILE

This will be needed especially if using a low end VPS

```
sudo fallocate -l 4G /swapfile && sudo chmod 600 /swapfile && sudo mkswap
/swapfile && sudo swapon /swapfile && echo '/swapfile none swap sw 0 0' |
sudo tee -a /etc/fstab && sudo sysctl vm.swappiness=10 && sudo sysctl
vm.vfs_cache_pressure=50 && echo 'vm.swappiness=10' | sudo tee -a
/etc/sysctl.conf && echo 'vm.vfs_cache_pressure=50' | sudo tee -a
/etc/sysctl.conf
```

[More information about swap files](#)

Set up FAIL2BAN

```
sudo service fail2ban restart
```

[More information about fail2ban](#)

Set up FIREWALL

Allow all ports needed (do this only if this is your first masternode install on this vps):

```
sudo ufw default allow outgoing && sudo ufw default deny incoming && sudo
ufw allow ssh/tcp && sudo ufw limit ssh/tcp && sudo ufw logging on &&
sudo ufw enable
```

Allow Sparks port

```
sudo ufw allow 12875/tcp comment Crowdcoin
```

Check status

```
sudo ufw status
```

You should see the following lines (maybe only one if IPv6 is not enabled on your VPS):

```
    12875/tcp          ALLOW        Anywhere          # Crowdcoin
    12875/tcp (v6)     ALLOW        Anywhere (v6)     # Crowdcoin
```

[More information about ufw firewall](#)

Install masternode binaries

Get latest binaries [from github](#). At the time of writing, latest version is v0.12.3.2. You should check on github and adapt the following commands with latest binaries reference.

```
wget
https://github.com/SparksReborn/sparkspay/releases/download/v0.12.3.2/spa
rkscore-0.12.3.2-linux64.tar.gz

tar zxvf sparkscore-0.12.3.2-linux64.tar.gz

mv sparkscore-0.12.3/bin/sparks{d,-cli} /usr/local/bin/

rm -r sparkscore-0.12.3*
```

Initialize files (do this for each masternode)

For security reasons, it is always better not to run any application as root user. So we create a user that will be running your masternode:

```
adduser crc-01 && adduser crc-01 sudo
```

This creates user "crc-01" with root privileges (be able to run root commands using the "sudo" prefix)

Switch to "crc-01" user

```
su – crc-01
```

Create the Crowdcoin directory

```
mkdir .crowdcoincore && cd . crowdcoincore
```

Get the following four values for your configuration file:

- <user> : type some random characters, only letters and numbers
- <password> : type some random characters, only letters and numbers, not the same as <user>
- <vps ip> : the IP of your VPS (looks something like this: 192.168.155.20)
- <private key> : in your wallet app, go to Tools > Debug console, and type "masternode genkey"

You may have already done the private key step when filling the masternode.conf file of your windows wallet app. If so, take the private key you put in that file as it must be the same.

Edit *crowdcoin.conf* file

```
nano crowdcoin.conf
```

then copy&paste the following :

```
rpcuser=<user>
rpcpassword=<password>
rpcallowip=127.0.0.1
rpcthreads=8
listen=1
server=1
daemon=1
staking=0
discover=1
maxconnections=16
masternode=1
printtodebuglog=0

bind=<vps ip>
rpcbind=<vps ip>
```

Wish to donate? Here's my crc address: **CcAhsK2fhdqeCmfj3uf6RxcHcMcamvss7s**

```
rpcallowip=<vps ip>
rpcconnect=<vps ip>

masternodeprivkey=<private key>

addnode=84.17.23.43:12875
addnode=18.220.138.90:12875
addnode=86.57.164.166:12875
addnode=86.57.164.146:12875
addnode=18.217.78.145:12875
addnode=23.92.30.230:12875
addnode=35.190.182.68:12875
addnode=80.209.236.4:12875
addnode=91.201.40.89:12875
addnode=80.211.42.70:12875
addnode=80.211.42.25:12875
```

*** END OF COPY PASTE ***

Start daemon

Now that you have everything set up, time to start the daemon for real. To speed up the synchronization of the blockchain, get a booststrap file and put it in the Sparks directory:

```
wget
https://github.com/SparksReborn/sparkspay/releases/download/bootstrap/bootstrap.dat
```

And finally launch the masternode daemon for good

```
crowdcoind
```

Wait about 20-30 minutes for your masternode to sync completely (or do the next step: *create monitoring script* to monitor closely your masternode).

Then you can start it from your windows wallet by going to Masternode tab and click Start MISSING. **Your 1000 crc collateral must be at least 15 confirmations old** before you start your masternode.

Create monitoring script

This step in not mandatory but will help greatly for monitoring your masternode.

First, go to your windows wallet, open masternodes.conf file where you entered the information for your masternode (not covered in this guide) and get the TxID and name you gave your masternode (MN-01 in the following lines, just use anything you want)

Go back to root user and create a script:

```
logout
```

```
cd /usr/local/bin/
```

```
nano crc-status.sh
```

Copy&paste the following:

Wish to donate? Here's my crc address: **CcAhsK2fhdqeCmfj3uf6RxcHcMcamvss7s**

```
crowdcoin-cli getinfo
crowdcoin-cli mnsync status
crowdcoin-cli masternode status
echo "MN-01:`crowdcoin-cli masternode list | grep <your TxID>`"
```

*** END OF COPY PASTE ***

Save the file (ctrl X), then:

```
chmod +x crc-status.sh
```

Go back to your masternode user and run the script you created:

```
su – crc-01
```

```
watch crc-status.sh
```

Here, "`watch`"-ing this script will let you monitor the details of your masternode's synchronization. You may exit the watch at any time with ctrl C. You should wait until you get to this point:

```
{
  "version": 120106,
  "protocolversion": 70206,
  "blocks": <total number of blocks>,
  "timeoffset": 0,
  "connections": 16,
  "proxy": "",
  "difficulty": 22.7406807988061,
  "testnet": false,
  "paytxfee": 0.00000000,
  "relayfee": 0.00010000,
  "errors": ""
}
{
  "AssetID": 999,
  "AssetName": "MASTERNODE_SYNC_FINISHED",
  "Attempt": 0,
  "IsBlockchainSynced": true,
  "IsMasternodeListSynced": true,
  "IsWinnersListSynced": true,
  "IsSynced": true,
  "IsFailed": false
}
{
  "vin":
"CTxIn(COutPoint(0000000000000000000000000000000000000000000000000000000000000000,
4294967295), coinbase )",
  "service": "<your ip>:8890",
  "status": "Not capable masternode: Masternode not in masternode list"
}
MN-01:
```

The `blocks` number will go up until it syncs completely with the blockchain. This is the longest part. You can see what number it needs to reach by hovering over the small **V** on the lower right of your windows wallet app

When `AssetID` is set to 999, your 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 need to "start MISSING" from your windows wallet. **Your 1000 crc collateral must be at least 15 confirmations old** before you start your masternode.

Doing this, the last part should look like this:

```
{
  "vin": "CTxIn(COutPoint(<your TxID>, 1), scriptSig=)",
  "service": "<your ip>:12875",
  "payee": "<your collateral address>",
  "status": "Masternode successfully started"
}
MN-01:  "<your TxID>-1": "PRE_ENABLED"
```

The `PRE_ENABLED` should turn into `ENABLED` after a few minutes.

Your masternode is now completely operational.

Install Sentinel

Sentinel is not strictly needed for payouts, but if you want to monitor your masternode easily with 3rd party services such as nodecheck.io, you will have to install it.

Get source

```
cd
git clone https://github.com/crowdcoinChain/sentinelLinux.git sentinel/
cd sentinel && rm sentinel.conf
```

You may modify some files to allow multiple instances of masternode on same vps using different IPs.

```
cd lib
```

Edit *crowdcoind.py*, line 18:

- replace `kwargs.get('host', '127.0.0.1')` by `kwargs.get('bind')`

```
nano crowdcoind.py
```

Edit *crowdcoin_config.py*, line 31:

- add "bind" in params : match = re.findall(r'rpc(**bind**|user|password|port)=(.*?)$', data, re.MULTILINE)

```
nano crowdcoin_config.py
```

Set up python environment

```
cd ~/sentinel
virtualenv venv && venv/bin/pip install -r requirements.txt
```

The following command tests various aspects of sentinel, it must return "23 passed tests". You may use this command as soon as you launched the masternode.

```
~/sentinel/venv/bin/py.test ./test
```

The following command sets up files and must return nothing. If it does, something went wrong. Only use this command after masternode is fully synched and activated with collateral.

```
~/sentinel/venv/bin/python bin/sentinel.py
```

Finally, create crontab entries as crc-01 user to restart masternode on reboot and wake sentinel every five minutes

```
crontab -e
```

Add these lines to the file.

```
@reboot /usr/local/bin/crowdcoind & >/dev/null 2>&1
```

```
* * * * * cd /home/crc-01/sentinel && ./venv/bin/python bin/sentinel.py
>/dev/null 2>&1
```

Voilà! All set.

For multiple masternodes on the same VPS, you must have a different IP each time. You must do the following steps for each masternode you want to install:

1. Initialize files (use another username:, crc-02, crc-03…
2. Start daemon
3. Create monitoring script (actually only add a line with new TxID)
4. Install Sentinel

Wish to donate? Here's my crc address: **CcAhsK2fhdqeCmfj3uf6RxcHcMcamvss7s**