V.0.2.0

Hardware Decentralization User Guide

16/11/2018

DAppNode

VERSIONS COVERED:

DNP_VPN : 0.1.19

DNP_ADMIN: 0.1.13

DNP_IPFS: 0.1.4

DNP_ETHCHAIN: 0.1.9

DNP_DAPPMANAGER: 0.1.16

DNP_ETHFORWARD:0.1.2

DNP_BIND: 0.1.5

TABLE OF CONTENTS

Introduction - What is DAppNode? **3**

The Desired Final User Story **4**

How can you install DAppNode? **6**

Minimum requirements 6
How to install DAppNode on your machine 6
How to install DAppNode from an ISO 6
Download the image 6
Burn the ISO to an USB (~8 min) 6
Install an Ubuntu distribution (~15 min) 7
Recommended setup 7
Installation via script 9
WARNING 9
Script Installation guide 10
Install DAppNode and its dependencies 10
Install DAppNode **10**

Connect to the VPN **11**

Enter your DAppNode! **13**

Welcome to DAppNode – The Admin UI **13**
Dashboard 13
Activity 15
Devices 15
Installer 15
Testnet nodes 16
Livepeer 16
Monero 16
Packages 17
System 17
BIND 18
VPN 18
ETHCHAIN 18

ETHFORWARD 18
IPFS 18
WAMP 19
DAPPMANAGER 19
ADMIN **19**

What can you do now with your DAppNode?? 20

VPN connection 20
Parity 20
MyCrypto 20
Metamask 21
IPFS 23
ENS resolution **23**

Advanced users Bonus!!! 24

Have your own packages in your DAppNode 24
How to decentralize wallet.ethereum.org with DAppNode **25**

Troubleshooting

31

I can´t connect to the VPN when first installing DAppNode 31
I can´t connect with another device to my DAppNode 31
My ETH node never ends up syncing 31
Ports that need to be opened **32**

1 Introduction - What is DAppNode?

Welcome to DAppNode!!!

If you are reading this, decentralization, privacy and security are important to you and we're glad for that, and even more for helping people like you run their own secured hardware towards that which contributes to a truly decentralized web.

DAppNode is an operating system that allows you to host decentralized apps in a truly decentralized way, eliminating the reliance on vulnerable third parties and adds an extra layer of incentivization that helps to spread adoption of the blockchain ecosystem, and most important, solves the problem of infrastructure centralization.

You can install DAppNode on any piece of hardware (see minimum requirements) to get easy censorship-resistant access to your favorite DApps, the ability to host your own nodes and share access with family and friends.

Our goal is to achieve "real decentralization" by creating a layer of personally owned hardware infrastructure that is easy to set up and provides secure access.

DAppNode connects users to the decentralized web and creates the infrastructure for DApps to run services 24x7 in a truly decentralized manner.

We also developed an SDK for DAppNodePackage developers and we expect to see many DApps with their own DNP that improve the user experience and decentralizes access. For example, it can be a DApp accelerator that allows the devices not having to synchronize the chain to start using it, like a cache system for the DAppNode users. It could be a prepared service for staking in any chain, it could be a client for other chains, or it could be a scrawl of a decentralized search engine, etc.

Within the DAppNode project we have a storefront that will allow any person to install and configure their own project in just minutes, facilitating the visibility and decentralization of its growth and expansion. We believe this is the beginning of a new model, a P2P economy that makes projects truly decentralized and accessible to anyone.

We envision a self-sustaining system where users support their favorite projects, by providing the service that the project needs and in return it provides the necessary income to amortize the costs of the DAppNode.

1.1 The Desired Final User Story

1. Vojtech has several cryptocurrencies he follows and is ready to stop using his laptop to run clients but doesn't want to have to pay a monthly subscription in fiat to some company he doesn't trust with his private keys.

2. He hears about DAppNode and decides to buy the top of the line DAppNode Server preloaded from one of the many certified independent vendors (though he could buy his own server and install the software himself).

3. The DAppNode Server is delivered to his front door, and when Vojtech opens the box he finds a 2 step instruction card: First, plug in the Server. Second, go to dappnode.org/initialize

4. On this website, there is an extremely simple and short walk-thru for Vojtech to set up his credentials and choose which clients to deploy.

5. By just checking a few boxes, Vojtech deployed his own VPN; his own local instance of MyCrypto.com; Ethereum, Bitcoin, Monero and Dogecoin Full nodes; a ZenCash Secure Node and DASH Master Node; he joined the IPFS Consortium and deployed a TrueBit Verifier and Livepeer node; and hosted Giveth, Swarm City, Aragon, Colony, and Decentraland Helper DApps to support these projects and to ensure that his interaction with these DApps cannot be censored and are fully trustless.

6. Vojtech then tells his friends and family that he is the admin for his own DAppNode and he is able to give them a link that sets them up with their own credentials and access to his DAppNode, and Vojtech can deploy any other DApps that he or his friends and family want to have hosted with a couple clicks. He has become the trusted gateway to the decentralized world for his entire community.

7. Vojtech ends up being able to pay off the cost of his DAppNode in 2 months with the profits made from his ZenCash, Swarm City, DASH and TrueBit Nodes. He also canceled his VPN and all of his Digital Ocean accounts because now he has his own server that he can host all of his services on. Without even trying, Vojtech has

a couple extra hundred dollars a month because he is running DAppNode on his own personal server.

We started working on this project mostly part-time in October 2017; now we have a team of 6 working full time. We have a fully functional MVP version that we have been sharing with the community. We expect to partner with hardware manufacturers once we have a few more iterations of the software completed.

We have been developing in an invitingly open source manner and now that the software is completed anyone can use their own hardware to run DAppNode on it.

This project is a community funded and fully open sourced project that has received support from Ethereum Foundation, Ethereum Community Fund, Eth Prize, Aragon NEST, GitCoin and private donors.

We are open to receive donations from any interested parties to help pay the developers working hard on the software development.

See our campaign on Giveth.io to make a donation:
https://beta.giveth.io/campaigns/5b44b198647f33526e67c262

Ready to jump in??? Here we go!

2 How can you install DAppNode ?

Think of DAppNode as an operating system. In fact, it is; our savvy team has taken an Ubuntu distribution and superpowered it to be your gateway to access the decentralized web.

2.1 Minimum requirements

You will need a dedicated machine to install it.

Ethereum is big! To cope with that, you'll need to have a powerful enough machine to keep up with Ethereum's rapidly splitting blocks.

That is why we recommend having at least 8gbs RAM and a SSD hard drive with at least 160 GBs (add capacity as you like, keep in mind that Ethereum is 130GBs at press time).

2.2 How to install Dappnode on your machine

Okay, so you're ready to join the real DAppers. Let us show you how to make that hardware sing the hardware decentralization song!

2.2.1 *How to install DAppNode from an ISO*

2.2.1.1 *Download the image*

Download the image from DAppNode-ubuntu-18.04-server-amd64_v0.1.3.iso or build it from source

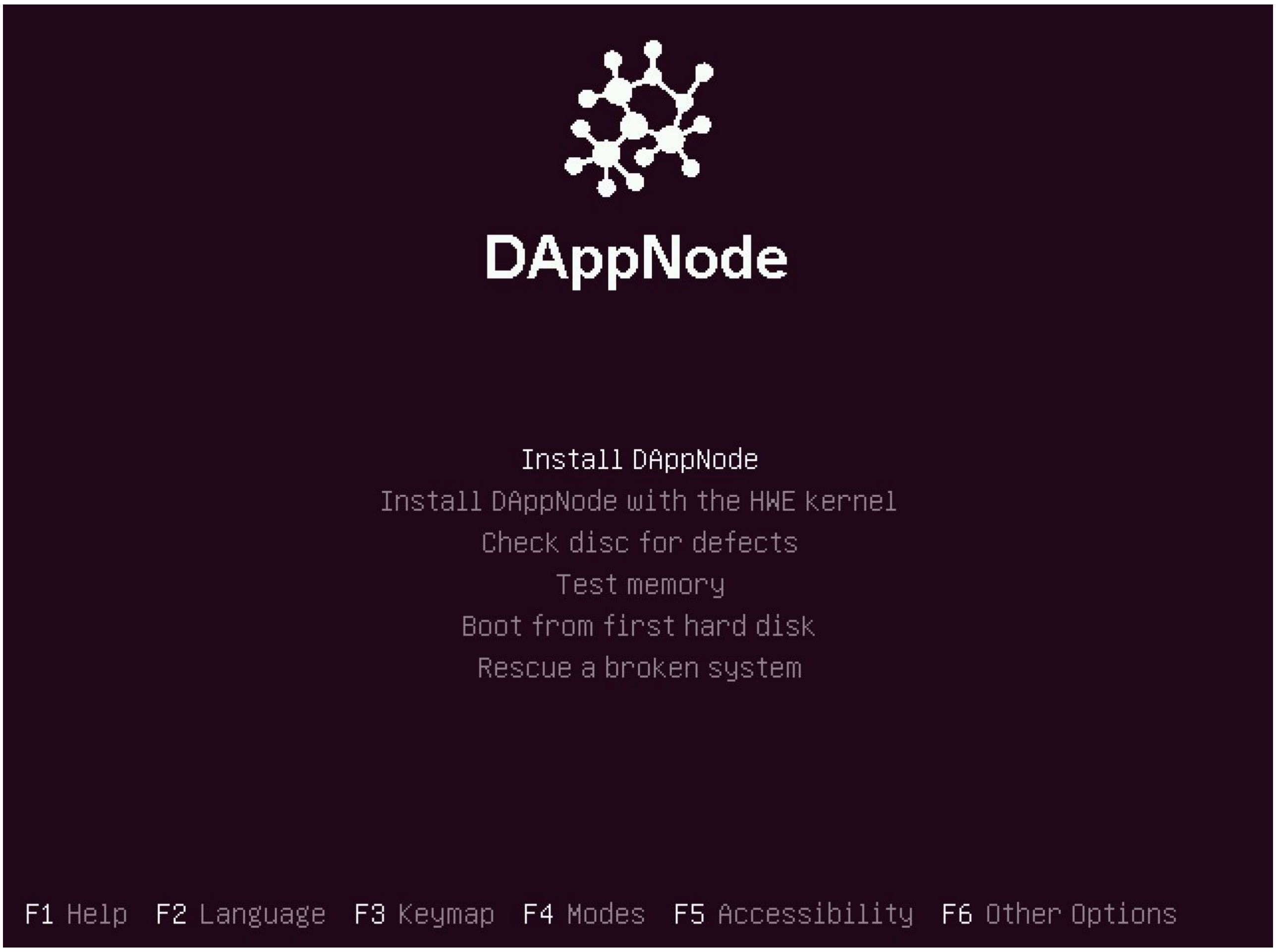
2.2.1.2 *Burn the ISO in a USB (~8 min)*

Burn the ISO to an USB stick, please follow the official Ubuntu instructions for MacOS, Windows or Ubuntu.

Please note that these instructions are intended to generate the ISO image in a bootable USB from a Mac, Windows or Linux device , and then install it in a server. **IF YOU EXECUTE THE BOOTABLE USB IN A MACHINE CONTAINING DATA IT WILL BE ERASED.** DAppnode is intended to run 24/7 so if you install it in a laptop or desktop machine and you turn it off it will lose the sync

2.2.1.3 *Install an Ubuntu distribution (~15 min)*

Insert the USB into your Server and prepare to install an Ubuntu distribution. You will have to make sure that your Server boots from the USB. If you succeed at booting up from your USB, you will be greeted with this screen or a similar one:



DAppNode

Install DAppNode
Install DAppNode with the HWE kernel
Check disc for defects
Test memory
Boot from first hard disk
Rescue a broken system

F1 Help F2 Language F3 Keymap F4 Modes F5 Accessibility F6 Other Options

Follow the Ubuntu installation steps; various screens will guide you through the process. You can follow this standard option for a default installation:

2.2.1.4 *Recommended setup*

Note by following these instructions you will erase your Server's disk contents.

1. **Select a language** - Language: [English]
2. **Select your location** - Country, territory or area: [United States]

3. **Configure the keyboard** - Detect keyboard layout? [Yes] Follow the instructions to detect your keyboard

4. **Configure the network** - Hostname: [DAppNode]

5. **Set up users and passwords** - Full name for the new user: [DAppNode]

6. **Set up users and passwords** - Username for your account: [dappnode]

7. **Set up users and passwords** Choose a password for the new user: [YOUR_PASSWORD]

8. **Set up users and passwords** - Re-enter password to verify: [YOUR_PASSWORD]

9. **Set up users and passwords** - Encrypt your home directory? [No]

10. **Configure the clock** - Is this time zone correct? [Yes]

11. **Partition disk** - Partitioning method: [Guided - use entire disk and set up LVM]

12. **Partition disk** - Select disk to partition: [SCSI33 (0,0,0) (sda) - ...]

13. **Partition disk** - Write the changes to disks and configure LVM? [Yes]

14. **Partition disk** - Amount of volume group to use for guided partitioning: [Continue]

15. **Partition disk** - Write the changes to disks? [Yes]

16. **Configure the package manager** - HTTP proxy information (black for none): [Continue]

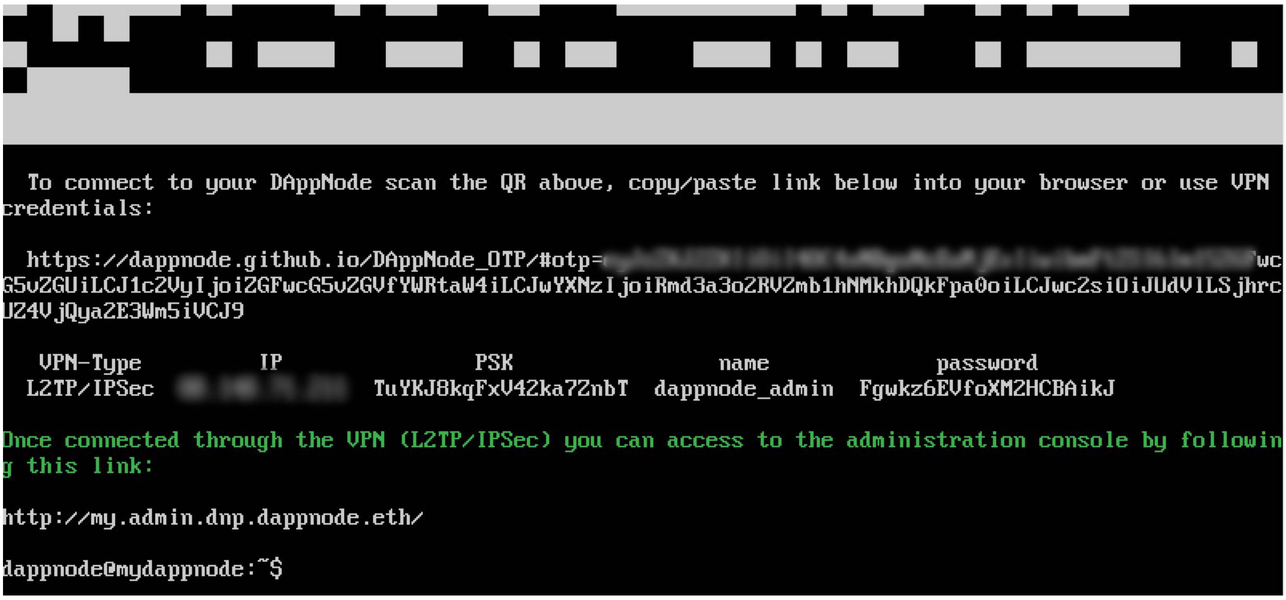
17. **Configuring tasksel** - How do you want to manage upgrades on this system? [Install security updates automatically]

18. **Software selection** - Choose software to install [OpenSSH server] Use the arrows to navigate to the option, and the spacebar to select it.

19. **Install the GRUB boot loader on a hard disk** - Install the GRUB boot loader to the master boot record? [Yes]

20. **Finish the installation** - Installation complete [continue]

If the installation succeeded, your system will reboot, you will have to log in with the user and password provided in the installation, and it should finish with this screen:



```
  To connect to your DAppNode scan the QR above, copy/paste link below into your browser or use VPN
credentials:

  https://dappnode.github.io/DAppNode_OTP/#otp=[redacted]wc
G5vZGUiLCJ1c2VyIjoiZGFwcG5vZGVfYWRtaW4iLCJwYXNzIjoiRmd3a3o2RVZmb1hNMkhDQkFpa0oiLCJwc2siOiJUdVlLSjhrc
UZ4VjQya2E3Wm5iVCJ9

   VPN-Type          IP                 PSK                   name              password
  L2TP/IPSec  [redacted]   TuYKJ8kqFxV42ka7ZnbT  dappnode_admin  Fgwkz6EVfoXM2HCBAikJ

Once connected through the VPN (L2TP/IPSec) you can access to the administration console by followin
g this link:

http://my.admin.dnp.dappnode.eth/

dappnode@mydappnode:~$
```

[Go to section 3: Connect to the VPN if this script installation stuff does not interest you…](#)

2.3 Installation via script

2.3.1 *WARNING*

This software is not meant to be run in a remote machine hosted by any remote provider. What DAppNode specifically wants to avoid is centralization of the machines that our digital lives rely on; nevertheless we understand that before buying a dedicated machine to run your Dappnode you might want to test it and see how easy it is to use….

And… only for that reason will we look aside when someone installs a DAppNode in a virtual provider. We want it to be clear that kind of use is not the recommended use, but for testing purposes only.

Remember: **Your hardware, your coins, your privacy, your freedom.**

2.3.2 Script installation guide

For this example we'll be installing DAppNode on a Digital Ocean droplet, but the process should work for any other Ubuntu Server 18.04.

We strongly recommend using 8GB+ of RAM and a 160Gb+ SSD hard drive.

2.3.2.1 Install DAppNode and its dependencies

Install the prerequisites (docker and docker-compose):

```
sudo wget -O -
https://github.com/dappnode/DAppNode/releases/download/v0.1.3/dappnode_insta
ll_pre.sh | sudo bash
```

2.3.2.2 Install DAppNode

```
sudo wget -O -
https://github.com/dappnode/DAppNode/releases/download/v0.1.3/dappnode_insta
ll.sh | sudo bash
```

If you have an static IP and want to set it up right from the connection, then run the command with the following variable including your staatic ip

```
sudo wget -O -
https://github.com/dappnode/DAppNode/releases/download/v0.1.3/dappnode_insta
ll.sh | sudo STATIC_IP="your static IP" bash
```

When the installation is done and is successful, you will be given credentials to connect to your DAppNode.

Take into account that the chain will take some time to synchronize and you will not be able to perform most of the actions before that.

3 Connect to the VPN

Once you have your DAppnode running, you will get an URL in your terminal from where you can download the VPN config file and install it on your device. Just download it and follow the instructions. For Android, Ubuntu and Windows you have to set the VPN following the instructions

Installing (or manually setting up)this VPN file will configure your VPN connection to your DAppNode from your device. The first device VPN connection will have super admin privileges so you can access and manage the DAppnode admin UI; this user cannot be deleted.

Take into account that some VPN clients might send all traffic through the VPN, which is not very ideal if you have many people connected to your DAppNode, or only to send traffic which points to an ETH domain.

DAppNode is not intended to manage all the traffic of the devices connected to it, only the ETH domains access requests.

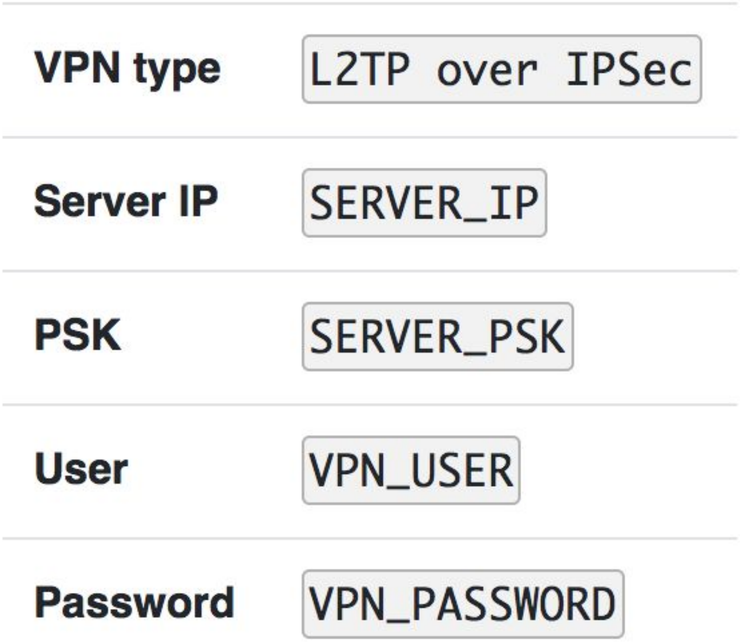
As a general rule, if your DAppNode is not connected through a router that supports UPnP, you will have to manually open some ports to have complete functionality. For the VPN to work you must make sure that you have opened the 4500 and 500 ports (UDP).

Here you have a sample table of the ports that should be opened in your DAppNode:

Service	**TCP**	**UDP**
VPN		500,4500
SSH	22	
Ethereum Node	30303	30303
IPFS	4001	4002

If you are not able to download or install the config file, you can set it up manually following the instructions for different platforms contained in the link that the terminal gives after installation. All the parameters you need to fill in are given by your terminal when you first install it or when you connect to it vía SSH.

These are the parameters you will need to configure, still depending on your operating system follow the instructions you will find in the website you will be directed to.



VPN type	L2TP over IPSec
Server IP	SERVER_IP
PSK	SERVER_PSK
User	VPN_USER
Password	VPN_PASSWORD

VPN TYPE: Select L2TP over IPSEC

Server IP: Select the IP given by the terminal when connecting to the DAppNode via SSH. If you are behind a router without NAT Loopback, you will also find in the terminal the internal IP you need to use to be able to connect being in the same network without NAT Loopback enabled.

PSK: Shared secret, you will find it in the terminal.

VPN User: This is the username of the super administrator and your terminal will give it to you too.

Password: You can also find the password in the terminal output when connecting to your DAppNode.

Now it is time…

4 Enter your DAppNode!

Navigate to my.admin.dnp.dappnode.eth to access DAppNode's administrative page. Bear in mind that DAppNode's functionality will be limited until the Ethereum mainnet chain is synced (should take around 2~3 hours to get a snapshot).

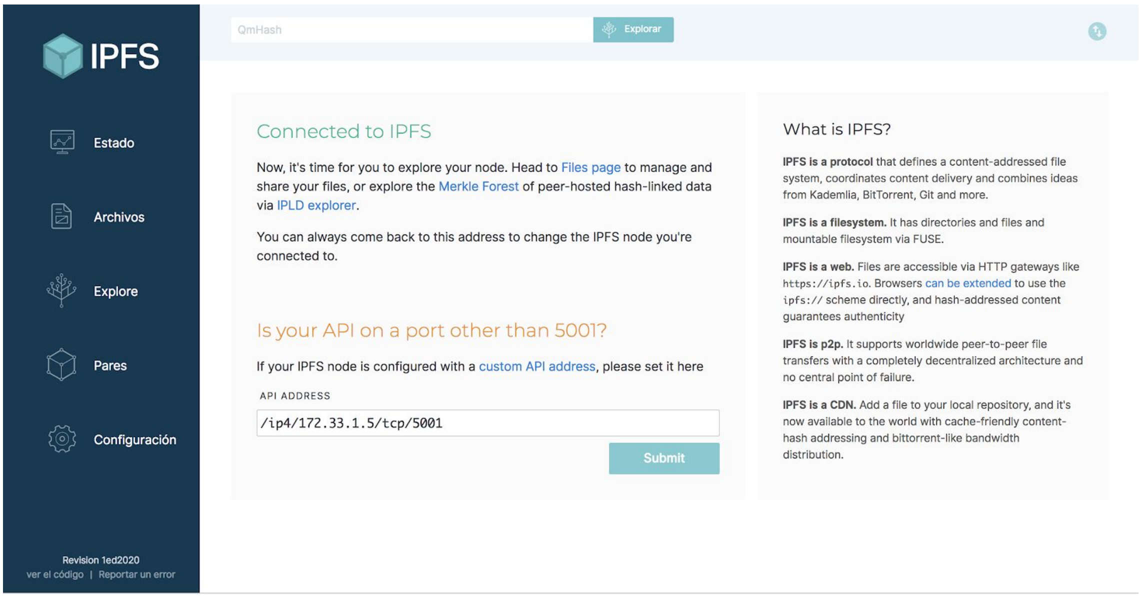
Now you can do things like for example:

- Navigate to a decentralized web decentral.eth:
  - Decentralized version of Mycrypto
  - Decentralized version of ENS Manager
  - Decentralized version of Wallet Gnosis
- Go to IPFS by entering http://my.ipfs.dnp.dappnode.eth:5001/webui into your browser.
- You have a websocket of your parity node in ws://my.ethchain.dnp.dappnode.eth:8546 and you can use http://my.ethchain.dnp.dappnode.eth:8545 as a custom RPC to connect to metamask i.e

NOTE ABOUT ACCESSING IPFS WEBUI:

We have updated our IPFS package (v.0.1.4), and one of the features is to provide a more complete and user friendly web interface. The first time you access to it will ask you for your "Custom API address", just fill the field with this address and you will be connected to your IPFS node , this is the input you have to enter in the field seen in the image below.

/ip4/172.33.1.5/tcp/5001



Connected to IPFS

Now, it's time for you to explore your node. Head to Files page to manage and share your files, or explore the Merkle Forest of peer-hosted hash-linked data via IPLD explorer.

You can always come back to this address to change the IPFS node you're connected to.

Is your API on a port other than 5001?

If your IPFS node is configured with a custom API address, please set it here

API ADDRESS

/ip4/172.33.1.5/tcp/5001

Submit

What is IPFS?

IPFS is a protocol that defines a content-addressed file system, coordinates content delivery and combines ideas from Kademlia, BitTorrent, Git and more.

IPFS is a filesystem. It has directories and files and mountable filesystem via FUSE.

IPFS is a web. Files are accessible via HTTP gateways like `https://ipfs.io`. Browsers can be extended to use the `ipfs://` scheme directly, and hash-addressed content guarantees authenticity

IPFS is p2p. It supports worldwide peer-to-peer file transfers with a completely decentralized architecture and no central point of failure.

IPFS is a CDN. Add a file to your local repository, and it's now available to the world with cache-friendly content-hash addressing and bittorrent-like bandwidth distribution.

5 Welcome to Dappnode – The Admin UI

Once you have succeeded in connecting to your DAppNode via VPN, you will have access to the above-mentioned URL. Be aware of this historic moment; it might very likely be the first ETH domain you visit.

The Admin UI allows the installation of packages in your DAppNode, adding devices to connect for your friends and family, monitoring its health and allowing fully functional operation without having to open the console.

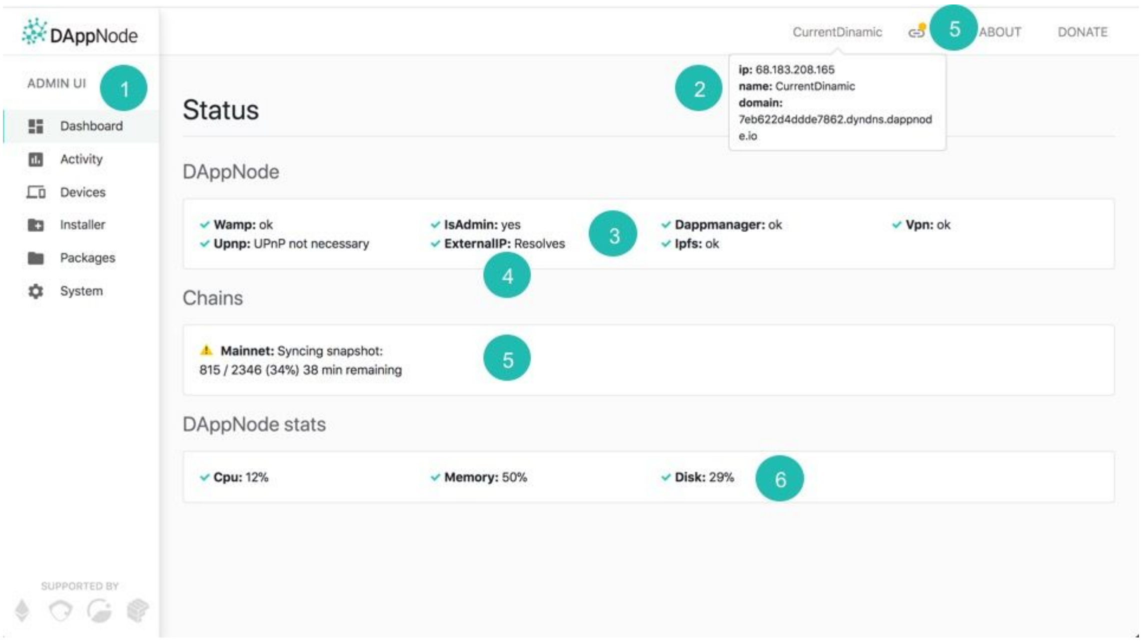
Let's have a look to see what you will find here.

5.1 Dashboard

When you first enter the Admin UI, you will see a link to a survey made to tell us how the installation went and provide your insights about the process. We greatly appreciate this feedback in order to help us to make a better product.

By clicking "Dashboard" you can see a summary of the components that make up the system and its status. This is what a healthy DAppNode dashboard looks like:

1 Tabs area
2 Name of the server / Internal IP
3 Health of core services
4 Error reporting area (in this example, the DAppNode is in good shape so nothing appears here)
5 Sync State
6. Server stats



ip: 68.183.208.165
name: CurrentDinamic
domain: 7eb622d4ddde7862.dyndns.dappnode.io

Status

DAppNode

✓ **Wamp:** ok
✓ **Upnp:** UPnP not necessary
✓ **IsAdmin:** yes
✓ **ExternalIP:** Resolves
✓ **Dappmanager:** ok
✓ **Ipfs:** ok
✓ **Vpn:** ok

Chains

⚠ **Mainnet:** Syncing snapshot: 815 / 2346 (34%) 38 min remaining

DAppNode stats

✓ **Cpu:** 12%
✓ **Memory:** 50%
✓ **Disk:** 29%

IMPORTANT NOTE ON SERVER STATS: if you note that the disk capacity is getting near to 100 % you should disable services to avoid the disk getting full. If this happens the DAppNode will get unfunctional, and you will not be able to erase packages once the memory is at 100%

5.2 Activity

This tab allows easy access to the DAppNode logs in order to debug errors. We work hard to make this tab irrelevant to you, but for the time being, if you are experiencing any issue, these logs will help our support team help you identify and fix any problem. You will also find a button to download a log report file.

5.3 Devices

This tab is one of the cornerstones of our vision, as the functionality is meant to build trusted circles that connect to the decentralized web through a DAppNode installed on a piece of self-owned hardware that provides access to your friends and family.

Just click “add a device”, name it and you will get a QR / link that contains the file to configure the VPN to connect to your DAppnode for any friend or family member.

This is the same process that you performed while installing your DAppNode when you accessed the first URL that the DAppNode sent you to. In fact, the DAppNode created the first device for you, now it is your turn…

Take into account that each device added has its own VPN credentials and is valid only for one concurrent connection, but you can add as many devices as you want.

If there is a device using the same VPN credentials, you will be able to connect to the VPN with other devices, but you won´t be able to access the DAppNode.

You also have the ability to give a device admin privileges so the ADMIN UI can be used by them. If any device without admin credentials tries to access the ADMIN UI, it will not work.

Take in account that if you remove admin privileges to any device while that device is connected to the server, it will still be able to access the admin UI and thus that device can make itself admin again, to prevent this, after removing admin privileges to any user you should restart the VPN package by going to System / VPN / Restart. This also applies when you want to remove access to any device without admin privileges.

5.3.1 Guest User functionality

In the devices section you will find a Guests functionality that can be enabled / disabled at your own discretion.

This functionality allows multiple users to connect with a single pair of credentials, what is specially useful in teaching environments or when you want to connect people without adding a device for each user.

When you disable this functionality, any user that is connected to the VPN will still be able to use your DAppNode until you restart the VPN service in System / VPN / Restart.

5.4 Installer

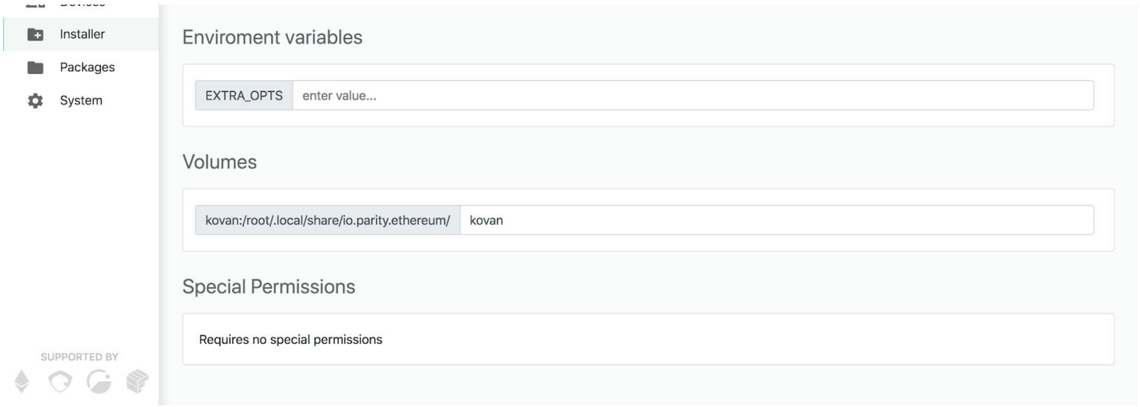
Here you have the DAppstore where you can directly install a growing amount of services and libraries just by a click and they will install. Please let us know which ones would you like to have in the DAppstore by filling out this little survey.

https://goo.gl/forms/EjVTHu6UBWBk60Z62

From the installer you can also install packages not shown in the interface by pasting its IPFS hash or its ENS domain in the above bar. The interface will show you the corresponding package to that IPFS hash/ETH domain and you will be able to install it if there are no compatibility issues. (see below section Have your own packages in DAppNode)

We have added a functionality that allows to customize some packages with predefined configurations made by the developer of the node/ DApp, please check the project documentation to see which options to customize are available.

You can also select your own customized path for the installation of the package by writing your selected path in the field aside the path by default



Enviroment variables

EXTRA_OPTS enter value...

Volumes

kovan:/root/.local/share/io.parity.ethereum/ kovan

Special Permissions

Requires no special permissions

Here you have a brief description of some of the available packages:

5.4.1 Testnet nodes

With DAppNode, you can easily set up nodes of the Rinkeby and Kovan networks for testing purposes. Just find in Packages the testnet you want to set up in your DAppNode, install it and it will immediately start to synchronize.

As with Parity Main net node, you have your websocket in the port 8546 and your RPC connection in the port 8545 using the following URLs:

my.rinkeby.dnp.dappnode.eth
my.kovan.dnp.dappnode.eth
my.ropsten.dnp.dappnode.eth

5.4.1.1 Görli:

Görli, the only Proof of Authority testnet that has compatibility with Geth, Pantheon and Parity is available for you to run with a couple of clicks in your DAppNode.

5.4.2 Livepeer

Livepeer is an open Source Video Infrastructure Services platform, built on the Ethereum Blockchain (Rinkeby). With the Livepeer package you can easily set up a Livepeer node. Have a look to https://livepeer.org to get more info and docs.

Please note that if you install this package that is running in Rinkeby, you need to have installed the Rinkeby chain for LivePeer to work properly. This is the same with any other services. You will need to have a node of the chain it is running on.

5.4.3 Swarm

Swarm is a distributed storage platform and content distribution service, a native base layer service of the ethereum web3 stack that we have made available as a DAppNode package so you can easily install and maintain your own Swarm node.

5.4.4 Monero

DappNode has a Monero daemon package available that will be your very own Monero node, as using Monero without your own node is a bit like having your DappNode in AWS (defeats the purpose).

Let's see how to connect a Monero wallet to your node.

Once you have installed the Monero daemon, with a couple of clicks you are ready to set up your wallet connected to your node.

In Monero a node is called a daemon, and does not have a wallet functionality. They are two separate pieces of software that work together by connecting your Graphic User Interface, or command line wallet to your own node.

For this example, we will use the official Monero GUI that you can get at www.getmonero.org

Select the GUI wallet version for your OS. Install it and open it. After showing you your keys and so on, the app will ask you which node you want to connect to.Simply select remote node, include http://my.monero.dnp.dappnode.eth as the node address, 18081 as the port and you are done!!!

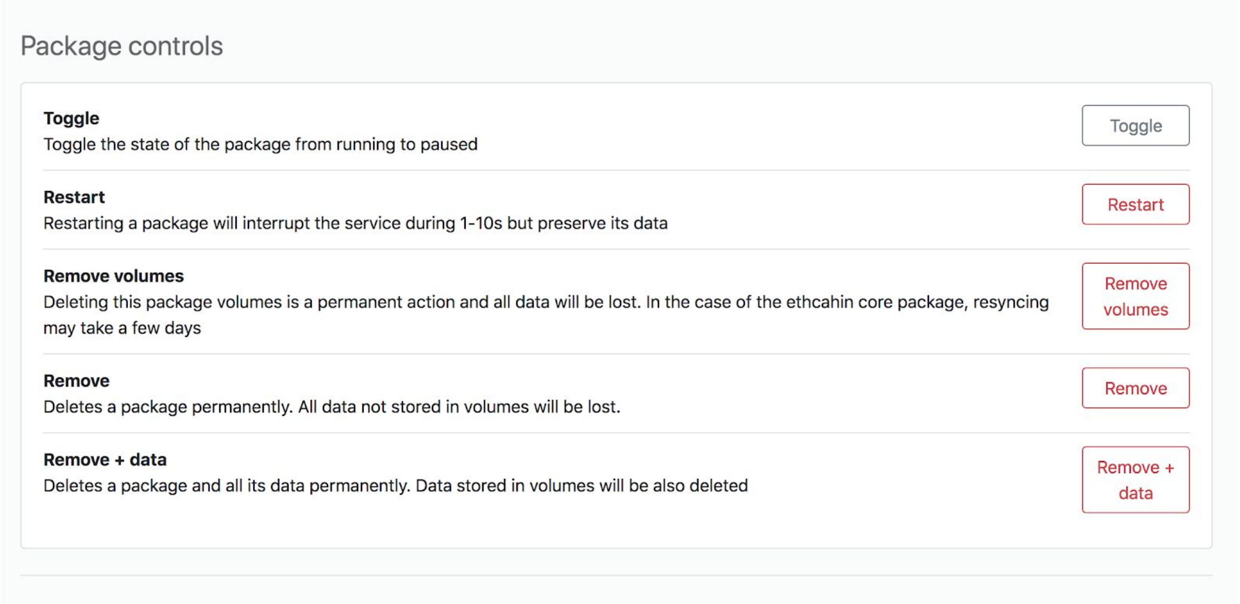
Now you have your Monero wallet connected to your own node!!!!

Do not buy any tanks please ;)...

5.5 Packages

Here you can see the packages you have installed and manage them, access to their logs, stop and restart them, remove them and preserve its data, or remove the package and the data.

These are the main options you can execute on your installed packages:



Package controls

Toggle
Toggle the state of the package from running to paused

Restart
Restarting a package will interrupt the service during 1-10s but preserve its data

Remove volumes
Deleting this package volumes is a permanent action and all data will be lost. In the case of the ethcahin core package, resyncing may take a few days

Remove
Deletes a package permanently. All data not stored in volumes will be lost.

Remove + data
Deletes a package and all its data permanently. Data stored in volumes will be also deleted

5.6 System

Here you can access the packages that are part of the DAppNode core and manage them, see their logs, restart them or delete its associated data to be restored.

If you have a Static IP you can set it up here so the future VPN credentials generated point to that fixed ip, just include your Static IP in the box and hit "Set", you can always disable.

Even though this might get a bit technical here for a user guide, here is a short description of each package:

5.6.1 BIND

Local DAppNode DNS. Links each package docker IP to a name in the format of my.[package-name].dnp.dappnode.eth. It also redirects .eth domains to the ethforward.

5.6.2 VPN

Provides a basic VPN for users to consume dappnode's services.

It runs a xl2tpd process alongside a nodejs app, both controlled by a supervisord process. The nodejs app connects with the WAMP to manage VPN users directly editing the /etc/ppp/chap-secrets file, which holds the user's credentials.

The user IP is static and set when that user is created. The static IP is used by the WAMP for authentication to allow only admin users to perform certain actions. Currently, there are three types of users:

- 172.33.10.1: Super admin. It is created when DAppNode is installed and can never be deleted
- 172.33.10.x: Admin user.
- 172.33.100.x: Non-admin user.

5.6.3 ETHCHAIN

Local full mainnet Ethereum node. Right now it uses Parity, but we are testing Geth against Parity to take a decision based on each client's efficiency, memory usage, and time to use among other parameters.

5.6.4 ETHFORWARD

Resolves .eth domains by intercepting outgoing requests, calling ENS, and redirecting to the local IPFS node.

It is a nodejs http proxy server, which also returns custom 404 pages if the content is not found or available or if the chain is still not synced.

5.6.5 IPFS

Local IPFS node. Its gateway is available at:

```
host: my.ipfs.dnp.dappnode.eth
port: 5001
protocol: http
```

5.6.6 WAMP

Handles inter-package communications. Restricts certain operations to only admin users.

We are using crossbar.io and its javascript client autobahn.js. Please refer to their documentation for more details.

5.6.7 DAPPMANAGER

Installs and manages DAppNode packages (DNPs). It's a nodejs app whose procedures are only consumed by the ADMIN, and depends on IPFS and ETHCHAIN to function.

5.6.8 ADMIN

Handles admin users <-> DAppNode interactions, such as managing packages or VPN users, is an NGINX process that serves a single-page React app that consumes RPCs of the DAPPMANAGER and the VPN.

6 What can you do now with your Dappnode??

6.1 VPN connection

Now you have your own VPN service to privately connect to your DAppNode and also to provide access for your family and friends to connect to ETH domains through your DAppNode.

6.2 Parity

After a few hours of installing DAppNode you will have your own Ethereum node running in your DAppNode.

You have available your parity websocket in ws://my.ethchain.dnp.dappnode.eth:8546 and RPC connection through http://my.ethchain.dnp.dappnode.eth:8545

6.2.1 MyCrypto

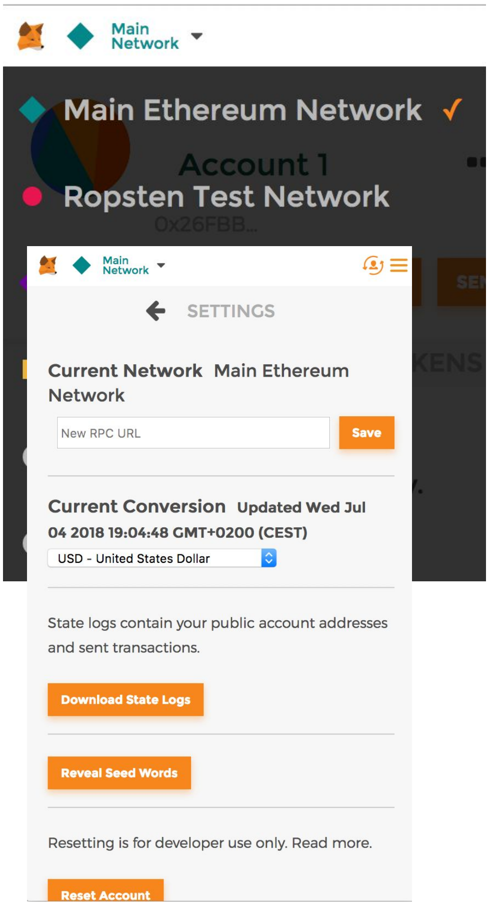
You can now enter a decentralized version of MyCrypto that it is not only hosted in IPFS but is using your node to connect to the Ethereum network. However, note that as the access URL is not https there might be incompatibilities; we are working hard to solve this issue and give you an awesome user experience using MyCrypto in a decentralized way. Just access it in http://mycrypto.dappnode.eth/ for now.

6.2.2 *Metamask*

You also can use Metamask connected to your own node, not with the pre-set Metamask nodes. To configure your Metamask while connected to your DAppNode, just follow these steps:

First, you must be connected to your DAppNode's VPN:

1. Click "Main network" on top to the left.



Main Network

Main Ethereum Network ✓

Account 1

Ropsten Test Network

0x26FBB...

Main Network

SETTINGS

Current Network Main Ethereum Network

New RPC URL

Save

Current Conversion Updated Wed Jul 04 2018 19:04:48 GMT+0200 (CEST)

USD - United States Dollar

State logs contain your public account addresses and sent transactions.

Download State Logs

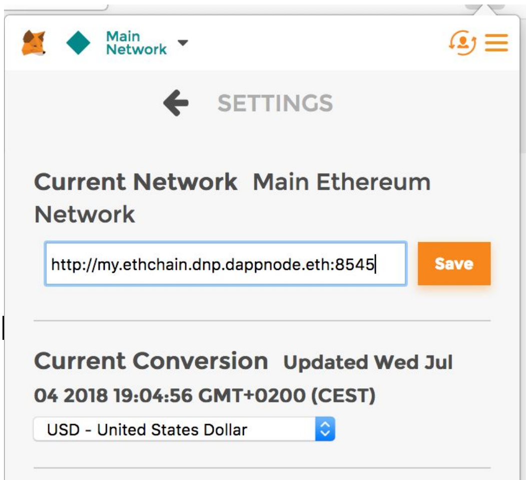
Reveal Seed Words

Resetting is for developer use only. Read more.

Reset Account

2. Click Custom RPC.

3. "New RPC URL": http://my.ethchain.dnp.dappnode.eth:8545



Main Network

SETTINGS

Current Network Main Ethereum Network

http://my.ethchain.dnp.dappnode.eth:8545

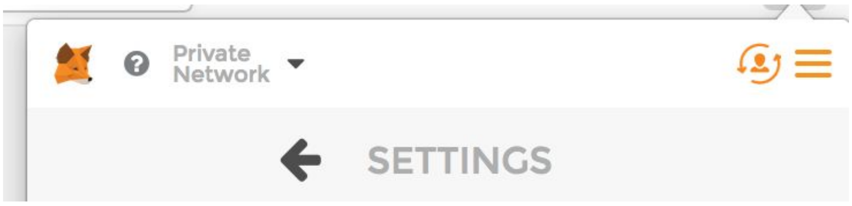
Save

Current Conversion Updated Wed Jul 04 2018 19:04:56 GMT+0200 (CEST)

USD - United States Dollar

"Al ne!!

4. Now you should be connected to "Private Network" and that's it!!



6.3 IPFS

When you install DAppNode an IPFS daemon is installed and your account is automatically created so you can start uploading and requesting the decentralized storage that the InterPlanetary File System offers.

You can access the web ui entering
http://my.ipfs.dnp.dappnode.eth:5001/webui

We have updated our IPFS package (v.0.1.4), and one of the features is to provide a more complete and user friendly web interfaz. The first time you access to it will ask you for your "Custom API address", just fill the field with this address and you will be connected to your IPFS node , this is the input you have to enter in the field seen in the image below.

/ip4/172.33.1.5/tcp/5001

If you want to know a bit more on IPFS here is a useful link:

https://medium.com/coinmonks/a-hands-on-introduction-to-ipfs-ee65b594937

6.4 ENS resolution

When your device is connected to a #DAppNode, you can use ".eth" domains that resolve to ipfs/swarm hashes.

Note that your browsing device is connected to your DAppNode via VPN, and the VPN is configured to distinguish DNS or ENS traffic, to send only the ENS traffic through the DAppNode (make sure in your VPN config that you are not sending all the traffic through the VPN). When you access a .eth domain from your browser, the DAppNode uses the ETHFORWARD core package to resolve the .eth domain to a IPFS hash via ENS, then the DAppNode looks for the hashed content in IPFS and serves the content to your browser.

Now you can seamlessly navigate ETH domains in a decentralized way.

7 Advanced users Bonus!!!

7.1 Have your own packages in your DappNode

You have two ways to install your own DNPs (DAppNode Packages):

1. With their ENS name, i.e. kovan.dnp.dappnode.eth (a private repo controlled by our team) or yourpackage.public.dappnode.eth, a repo with free access is available so anyone can publish packages there.

Ultimately any ENS name that resolves to an IPFS containing a valid manifest is acceptable.

2. With an IPFS hash that contains a valid manifest, i.e. /ipfs/QmR28vMrQVkSLznCQF7G1eNPx3RBx27zKYpctwwgRAot9W

So for development what we recommend is:

1. Develop your package and test it locally
2. Use the SDK to build it and upload it to IPFS
3. Test the package by installing it with its IPFS link
4. Once you are sure it works perfect, publish it to the open repo public.dappnode.eth.

Please take a look at these refs regarding the SDK to deploy your own packages in your DAppnode:

https://github.com/dappnode/DAppNodeSDK

https://github.com/dappnode/DAppNode/wiki/DAppNode-Package-Development

https://github./dappnode/DAppNode/wiki/DAppNode-packages-manifest

7.2 How to decentralize wallet.ethereum.org with DAppNode

In this section, we're going to explain how it's possible to decentralize wallet.ethereum.org using DAppNode and by decentralize we mean; distribute it's content through IPFS, making it possible to resolve via an ENS address.

Our first step is to clone the wallet.ethereum.org repository and follow the steps of the official guide, here (this tutorial is based on this commit 6a6463b1a6aa615e4364592c12c933ee816fb28b).

Here's a summary of the steps that we will need to follow:

```
# Install Meteor:


$ curl https://install.meteor.com/ | sh
$ git clone https://github.com/ethereum/meteor-dapp-wallet.git
$ cd meteor-dapp-wallet/app
$ npm install
$ npm install -g meteor-build-client
$ meteor-build-client ../build - path ""
```

After following these steps we will have a folder called `build` with a deployable version of wallet.ethereum.org.

> In order for us to use the web inside DAppNode a site must be a static website that can be used directly from IPFS (another case would be a website with an accelerator or a DAppNode Package API installed to act as a backend). If we must make these modifications, the modifications will depend on each website and how it has been developed.

- meteor-dapp-wallet/app/client/lib/ethereum/1_web3Init.js

Change:

```
web3 = new Web3('ws://localhost:8546');
```

By:

```
web3 = new Web3('ws://my.ethchain.dnp.dappnode.eth:8546');
```

After this step, the wallet will connect directly to our DAppNode, in case of not being able to go through MetaMask.

- meteor-dapp-wallet/app/client/lib/ethereum/observeTransactions.js

In line 532 change:

```
Session.get('network') == 'main' &&
```

By:

```
Session.get('network') == 'centralized' && false &&
```

The reason we need to make this change is that we do not want the website to connect to centralized services like min-api.cryptocompare.com. Making this change means that we lose the functionality which converts the eth to other currencies, in future versions this service should be replaced by a decentralized service to get the exchange rate.

- meteor-dapp-wallet/app/client/lib/ethereum/walletConnector.js

There seems to be a problem with the EthAccounts.init so we have to wrap line 86

```
try {

   EthAccounts.init();

} catch (err) {

   console.log(err);

}
```

Then comment `EthTools` on line 89, to avoid min-api.cryptocompare.com calls

```
/*

   EthTools.ticker.start({

      extraParams:  typeof  mist  !==  'undefined'  ?  'Mist-'  +
mist.version  :  '',currencies:  ['BTC',  'USD',  'EUR',  'BRL',
'GBP']   });

*/
```

- meteor-dapp-wallet/app/client/routes.js

To redirect on start to dashboard modify line 14:

```
if (location.origin === 'file://') {
```

To:

```
if  (location.origin  ===  'file://'  ||  location.origin  ===
'http://my.ipfs.dnp.dappnode.eth:8080') {
```

- Solve SourceSansPro-ExtraLightIt error

```
wget                                                         -O
../build/packages/ethereum_dapp-styles/fonts/SourceSansPro-Extr
aLightIt.otf
https://www.wfonts.com/download/data/2015/10/11/source-sans-pro
-extralight/Source%20Sans%20Pro%20ExtraLight%20Italic.otf
```

Once these steps are finished we're in a position to create a new build:

```
$ meteor-build-client ../build — path ""
```

Upload to IPFS

When uploading content to IPFS we use a tool we have created that can be used if you are connected to DAppNode (this tool is currently experimental, and will be improved in the coming months) or you could use the command `ipfs add -r build`.

Using our tool the next steps would be:

```
$cd ..
```

```
$npm install -g @dappnode/ipfsuploader
```

```
$ipfsuploader build/
```

After executing the last command you will obtain a similar output (they will not be the same hashes) to this one:

…

```
Qmb5oxJWf5Zw1UnvewkRM6V5qVbxWcY5s59FvhtWhC6F4F build/i18n


QmQV1tXNCZsD82LiLwWpHvdwWqbGXJd8q1Pq2hMPxyiKFa
build/packages/es5-shim


QmTte2i1HQKRgUgA8ZuVANwmqLCjjYzddmCekbUqJ3fmCA
build/packages/ethereum_dapp-styles/fonts


QmWMVompWymG8CmCgyB57dvaWegymjknMFTUQVrWfYebYu
build/packages/ethereum_dapp-styles/icons


QmTCXm13p6PW7CnKegNqFP3mCgt8sAaNNteCJDjFiGP3Jm
build/packages/ethereum_dapp-styles


QmVvCPByChGfmEvxS2Nv6icKZSJ27aYqDSUzP2gta44XYb
build/packages/ethereum_elements


QmXQ6fGzJsDAUGuLFxG8wgMgvyvgnR6pW9yeue3VUtdHne build/packages


QmTRpmNWiAkYQnesiGZRVE9NwbEfqZLH4DnLmbCmjMGaLL build/sockjs


QmZQ3GzqXHCRM6uccP6TcZdPGPSyqJXyhwLETD2T2o8m73 build
```

If we use the hash associated with `build` and access it through this URL: http://my.ipfs.dnp.dappnode.eth:8080/ipfs/QmZQ3GzqXHCRM6uccP6TcZdPGPSyqJXyhwLETD2T2o8m73

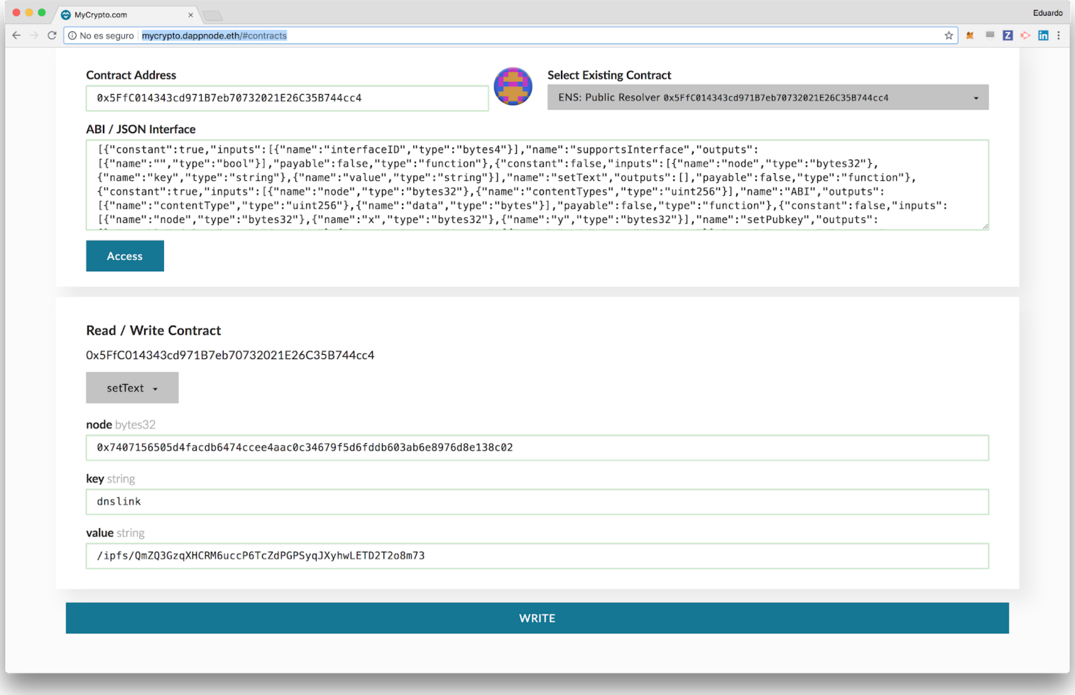
The website is now distributed in IPFS!

Point the ENS domain to the IPFS hash

If you are the owner of an ENS domain you can make this point to the hash you want. We are going to use wallet.dappnode.eth for this example:

1. Go to http://mycrypto.dappnode.eth/#contracts (if you don't have access to a DAppNode you can use the centralized alternative) https://mycrypto.com#contracts or https://www.myetherwallet.com/#contracts)
2. Select: ENS: Public Resolver 0x5FfC014343cd971B7eb70732021E26C35B744cc4
3. Access
4. Go to https://etherscan.io/enslookup and search for wallet.dappnode.eth noting its NameHash (in this case 0x7407….8c02)
5. setText

   - node bytes 32: 0x7407156505d4facdb6474ccee4aac0c34679f5d6fddb603ab6e8976d8e138c02
   - key: dnslink
   - value:/ipfs/QmZQ3GzqXHCRM6uccP6TcZdPGPSyqJXyhwLETD2T2o8m73



Contract Address

0x5FfC014343cd971B7eb70732021E26C35B744cc4

Select Existing Contract

ENS: Public Resolver 0x5FfC014343cd971B7eb70732021E26C35B744cc4

ABI / JSON Interface

```
[{"constant":true,"inputs":[{"name":"interfaceID","type":"bytes4"}],"name":"supportsInterface","outputs":
[{"name":"","type":"bool"}],"payable":false,"type":"function"},{"constant":false,"inputs":[{"name":"node","type":"bytes32"},
{"name":"key","type":"string"},{"name":"value","type":"string"}],"name":"setText","outputs":[],"payable":false,"type":"function"},
{"constant":true,"inputs":[{"name":"node","type":"bytes32"},{"name":"contentTypes","type":"uint256"}],"name":"ABI","outputs":
[{"name":"contentType","type":"uint256"},{"name":"data","type":"bytes"}],"payable":false,"type":"function"},{"constant":false,"inputs":
[{"name":"node","type":"bytes32"},{"name":"x","type":"bytes32"},{"name":"y","type":"bytes32"}],"name":"setPubkey","outputs":
```

Read / Write Contract

0x5FfC014343cd971B7eb70732021E26C35B744cc4

setText

node bytes32

0x7407156505d4facdb6474ccee4aac0c34679f5d6fddb603ab6e8976d8e138c02

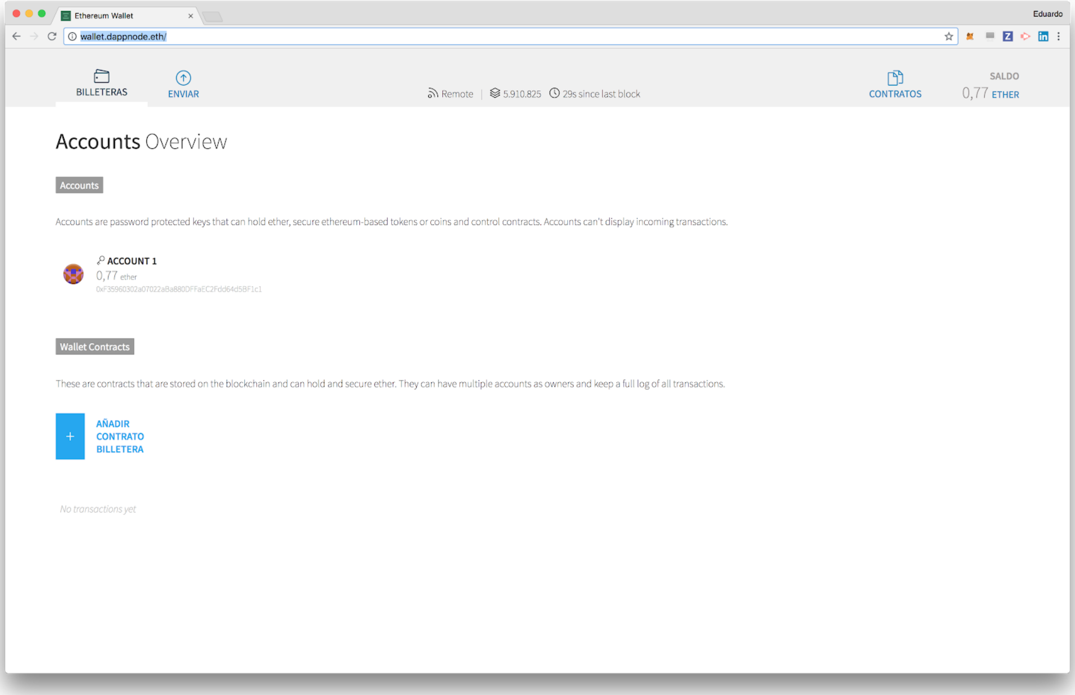
key string

dnslink

value string

/ipfs/QmZQ3GzqXHCRM6uccP6TcZdPGPSyqJXyhwLETD2T2o8m73

With these parameters we make the transaction in ethereum and once it's mined the web will be accessible from that domain!!



Accounts Overview

Accounts

Accounts are password protected keys that can hold ether, secure ethereum-based tokens or coins and control contracts. Accounts can't display incoming transactions.

ACCOUNT 1
0,77 ether
0xF35960302a07022aBa880DFFaEC2Fdd64d5BF1c1

Wallet Contracts

These are contracts that are stored on the blockchain and can hold and secure ether. They can have multiple accounts as owners and keep a full log of all transactions.

No transactions yet

8 Troubleshooting

8.1 I can´t connect to the VPN when first installing DappNode

If the device with which you are trying to connect to your DAppNode is in the same local network as the DAppNode, you should have a router that supports NAT Loopback or edit the VPN configuration and change the Server IP to the internal IP of your DAppNode that you will find in the admin UI Dashboard, and in the console when you connect to the server via SSH.

You can easily know if your router has NAT loopback enabled without accessing the router admin UI by opening your terminal and type ping + the IP of your DAppNode (where your VPN connection profile points to).

If the ping works without getting stuck your router has NAT Loopback so if you can´t connect to the VPN the cause is not being in the same network. Check that UDP ports 500 and 4500 are opened.

8.2 I can´t connect with another device to my DAppNode

Only one device can access your DAppNode for each VPN credentials created; each device connected should have its own VPN configuration adding the desired devices in the Devices tab of your admin UI. You can add as many devices as you want! Be nice and provide all your friends and family an access to the decentralized web please!!

8.3 My ETH node never ends up syncing

If you are not on an 8gb Ram configuration/your HD has not a high writing speed (ideally SSD), it might happen that the server is not able to catch up with Ethereum Blocks, so it never gets synced. We are sorry but the server might not be able to cope up with the chain.

8.4 Ports that need to be opened

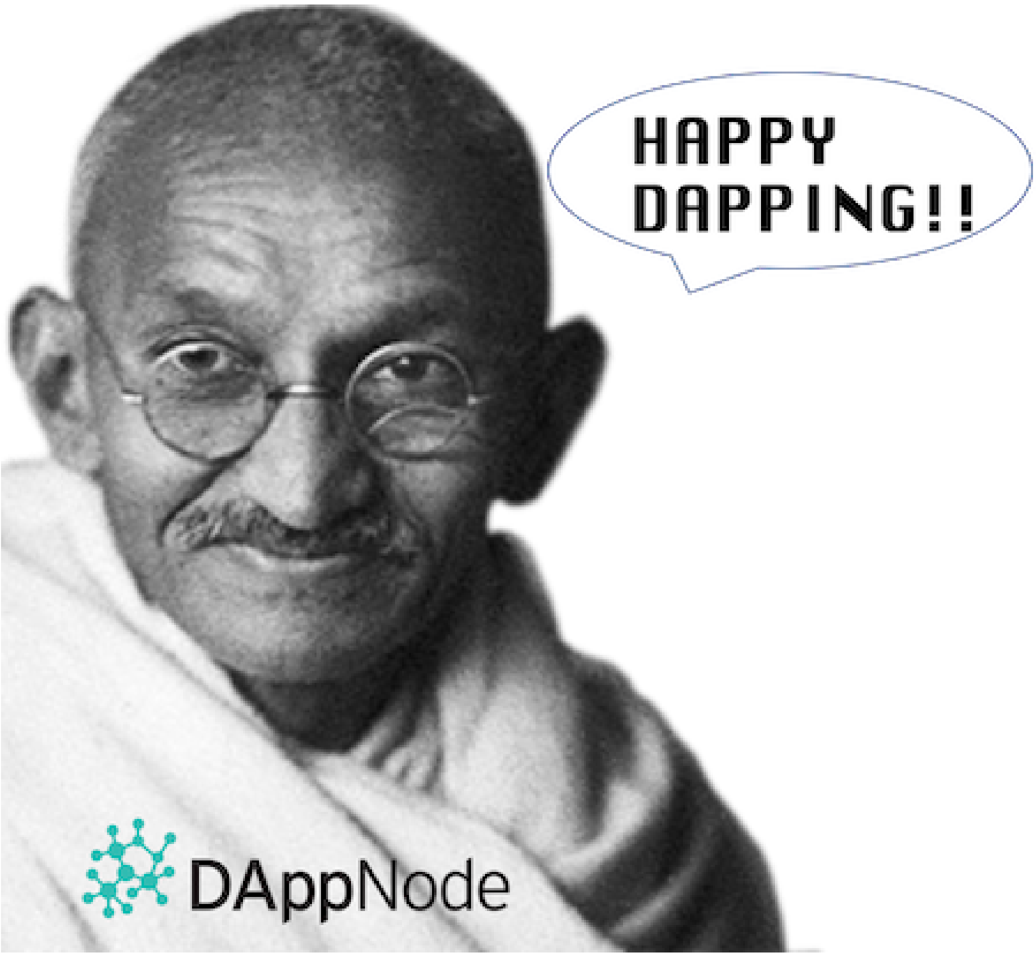
Please find in this table the ports that need to be opened for the smooth functioning of your DAppnode and installed packages.

Service	TCP	UDP
VPN		500,4500
SSH	22	
Ethereum Node	30303	30303
IPFS	4001	4002

However, if your router supports UPnP, do not worry about this, it will manage all the ports stuff for you.

10. LICENSE

DAppnode is licensed under the GNU General Public License V3. Click here to see the license


HAPPY
DAPPING!!
DAppNode