//for whatever device you are on you may need to make sure that ports 80, 8080, 4001, 5001, & 5002 are open. Installing ipfs from official binary distros (using this because I'm running on a non-gui instance). Instructions from <https://docs.ipfs.tech/install/command-line/#system-requirements>. Most of these instructions come from <https://blog.stacktical.com/ipfs/gateway/dapp/2019/09/21/ipfs-server-google-cloud-platform.html>, but there are some outdated instructions - i.e. using ipfs-update instead of kubo & some instructions I didn't follow due to my needs i.e. not making the ipfs service accessible to all users. Also used these instructions for some more clarity when it comes to DNS records & using nginx <https://gist.github.com/NatoBoram/09d244ab02af16fecb62b917f7bee3c0>

* wget https://dist.ipfs.tech/kubo/v0.24.0/kubo\_v0.24.0\_linux-amd64.tar.gz
* tar -xvzf kubo\_v0.24.0\_linux-amd64.tar.gz
* cd kubo
* sudo bash install.sh
* ipfs --version
* ipfs init

//set up IPFS service

* sudo bash -c 'cat >/lib/systemd/system/ipfs.service <<EOL

[Unit]

Description=IPFS Daemon

After=network.target

[Service]

ExecStart=/usr/local/bin/ipfs daemon

User=bgoss //I am not making this system-wide

Restart=always

LimitNOFILE=10240

Environment="IPFS\_PATH=/home/bgoss/.ipfs" //or wherever your .ipfs folder is

[Install]

WantedBy=multi-user.target

EOL'

//enable, start & check new ipfs service

* sudo systemctl daemon-reload
* sudo systemctl enable ipfs
* sudo systemctl start ipfs
* sudo systemctl status ipfs

//Configuring CORS on IPFS

* ipfs config --json API.HTTPHeaders.Access-Control-Allow-Origin '["<your domain or all (\*)>"]'
* ipfs config --json API.HTTPHeaders.Access-Control-Allow-Methods '["GET", "POST"]'

//go to .ipfs/config & double check that the following settings exist under the API & Addresses sections

- - API

"HTTPHeaders": {

"Access-Control-Allow-Methods": [

"GET",

"POST"

],

"Access-Control-Allow-Origin": [

"project5700-dist.store"

]

}

},

//this allows any ip address to access this port

- - Addresses

"Gateway": "/ip4/0.0.0.0/tcp/8080"

//these changes are the bare minimum. If you want to customize your gateway further here is some relevant information: https://github.com/ipfs/kubo/blob/master/docs/config.md#gateway-recipes

//after you have made the changes you want to

* sudo systemctl restart ipfs
* sudo systemctl status ipfs

//get your website set up

* ipfs get <CID>
* (orrr if you don't have pinned content yet: add it, pin it, and make sure it's pinned)
* ipfs add <folder> //this folder holds your .html
* ipfs pin add <CID>
* ipfs pin ls <CID>

//test CORS && Connectivity

* curl -v -H "Origin: http://<wrong or right domains>" \
* -H "Access-Control-Request-Method: POST" \
* -H "Access-Control-Request-Headers: X-Requested-With" \
* http://0.0.0.0:5001/api/v0/swarm/peers

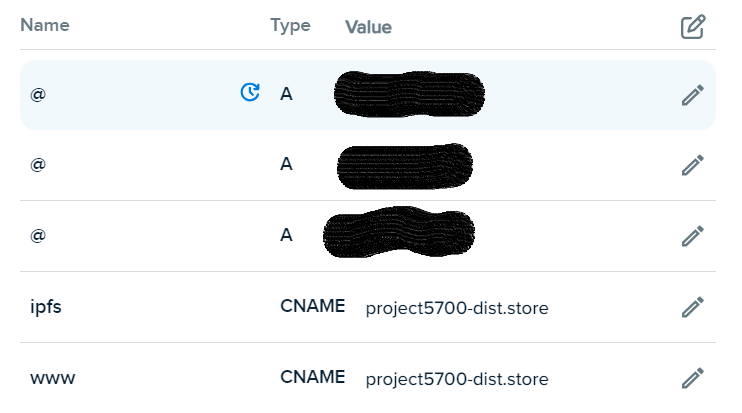
//its ok if you see "NOT ALLOWED" for the methods. The output should reflect how you want this set up to behave

//install nginx to set up a proxy server

* sudo apt update
* sudo apt install -y nginx
* sudo systemctl status nginx

//before you move onto the next steps you need to make sure that your ip address is in the dns records of your hostname. If you don't have any of this set up, set it up now.

//I used dreamhost but you can use whatever dns record holder you want. I did not pay for hosting, I just paid for the DNS rights.

// This is what my setup looked like The value in the blacked out areas are IP addresses. 

// it looks this way because I am attempting to host the website on three different instances. so i'm skipping the cert for the third instance. Certification didn’t work for my third instance, but I was able to get the first two verified. I think it has something to do with the multiple IPs pointing to the same hostname

//install ssl certificate using certbot: this is the site I used because I was on ubuntu, but it might be different for you <https://certbot.eff.org/instructions?ws=nginx&os=ubuntufocal>. install the cert only because you are going to handle your own configuration. Make sure you have snapd & don't have any other certbot items

* sudo snap install --classic certbot
* sudo ln -s /snap/bin/certbot /usr/bin/certbot
* sudo certbot certonly --nginx

//setting up nginx proxy to work with ipfs

* sudo cp /etc/nginx/sites-available/default /etc/nginx/sites-available/ipfs
* sudo vim /etc/nginx/sites-available/ipfs

//add the following & get rid of default config

server {

server\_name ipfs.example.org

location /ipfs {

proxy\_pass http://127.0.0.1:8080;

proxy\_set\_header Host example.org;

}

}

server {

server\_name \*.ipfs.example.org

location / {

proxy\_pass http://127.0.0.1:8080;

proxy\_set\_header Host $host;

}

}

//attempt to access the content through the ip address or hostname

~~ Connecting to Ethereum storage

//used these instructions to guide my next steps

<https://www.quicknode.com/guides/ethereum-development/dapps/how-to-integrate-ipfs-with-ethereum#what-is-ipfs>

