PoA EVM chain build and launch

1. installation
2. gvm install

apt update

sudo apt-get install bison

bash < <(curl -s -S -L <https://raw.githubusercontent.com/moovweb/gvm/master/binscripts/gvm-installer>)

apt-get install gcc

apt-get install make

apt upgrade

apt install golang-go

gvm install go1.20

gvm use go1.20 --default

- go 1.20/1.21

- go-ethereum1.13.15

- docker

- make

1. command

- make all/make geth

- create new account

./build/bin/geth account new

- create genesis.json

Set clique mode(period and epoch)

Set chain id

Set alloc account

Set extradata

- create genesis block

geth init genesis.json

// Main node

BOOTNODES="enode://78f1d2e54156b3516c87c157340ee64e805a33616312de4cbcc67174ec806f067c3276b8e3d3c28f823f5339a778a407478d9fb848eb02d0140d571f622ce3f6@172.234.21.63:30305,enode://04072a491cc95e377de07fea0aa9d4182baeb538f28172e0f8c16303170db606e65e1bf3b663421a6d4492b8ce2a1eed0a57b5e059e948a146b989d5d6541856@96.126.110.123:30305"

geth --http --http.api eth,net,web3 --port 30305 --bootnodes $BOOTNODES --unlock 0x1AeB89Fcf0B9c3464747bd23C39290fC86b563A9 --password ./config/password.txt --mine --miner.etherbase 0x1AeB89Fcf0B9c3464747bd23C39290fC86b563A9 --allow-insecure-unlock --http.addr 0.0.0.0

BOOTNODES="enode://a918f03a4080b60eaadc5463f1e0261259b86e1353f7f27bf18eb03ed63947de02b47ec10be74648d2b0995c13b2fa365879bd27094d30a8bd687f4fea6637c1@172.234.21.63:30305,enode://75403ce7414578adbbb43a4fe53b4c573c110440218a1a8bcc0453e877477afa836a51d2363b6ad9fa99f270123a49dc0600139dc7b9a8e708d1155d985a50d5@96.126.110.123:30305"

geth --http --http.api eth,net,web3,txpool --port 30305 --bootnodes $BOOTNODES --unlock 0x37a186e4Cc6302ab55C59e49C8E0eD742267Aab8 --password ./config/password.txt --mine --miner.etherbase 0x37a186e4Cc6302ab55C59e49C8E0eD742267Aab8 --allow-insecure-unlock

// Peer node

geth --http --http.api eth,net,web3 --bootnodes enode://9e4a5a7f0b236f8f64769109f229e4ae76f7f66107361c8b5ee82cacab3f372d800f87843a7cfa10f00d5a2cf01e15a823d51fdf28a27865afff5540a8a4dfe8@172.234.21.63:0?discport=30305 --networkid 20441

1. admin function
2. backend api

* install postgresql

sudo apt update

sudo apt install postgresql postgresql-contrib

sudo systemctl status postgresql

sudo systemctl start postgresql

sudo systemctl enable postgresql

**create postgresql user**

sudo -u postgres psql

CREATE USER admin\_backend WITH PASSWORD 'xusd\_admin';

\l

CREATE DATABASE xusd;

GRANT ALL PRIVILEGES ON DATABASE xusd TO admin;

\du

psql -h localhost -U admin -d xusd

main url: https://xusd/api

* whitelist
  + - add

/whitelist/add

* + - remove

/whitelist/remove

* blacklist
  + - add

/blacklist/add

* + - remove

/blacklist/remove

* excludecheckprice
  + - add

/excludecheckprice/add

* + - remove

/excludecheckprice /remove

* bmp
  + - add

/bmp/add

* + - remove

/bmp/remove

* + - updatefee

/bmp/updatefee

* knownexchange
  + - add

/knownexchange/add

* + - remove
    - /knownexchange/remove
* assetholder
  + - add

/assetholder/add

* + - remove

/assetholder/remove

go mod init your\_module\_name

go get github.com/lib/pq

go mod tidy

go run main.go

1. Docker Install

* Docker Install
* sudo apt update
* sudo apt install apt-transport-https curl gnupg-agent ca-certificates software-properties-common -y
* curl -fsSL <https://download.docker.com/linux/ubuntu/gpg> | sudo apt-key add -
* sudo add-apt-repository "deb [arch=amd64] <https://download.docker.com/linux/ubuntu> focal stable"
* sudo apt install docker-ce docker-ce-cli containerd.io -y
* apt install podman-docker
* apt install docker.io

Docker Composer 2.x.x install

* sudo apt update
* sudo apt install ca-certificates curl
* sudo install -m 0755 -d /etc/apt/keyrings
* sudo curl -fsSL [https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc](https://download.docker.com/linux/ubuntu/gpg%20-o%20/etc/apt/keyrings/docker.asc)
* sudo chmod a+r /etc/apt/keyrings/docker.asc
* echo \
* "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu \
* $(. /etc/os-release && echo "$VERSION\_CODENAME") stable" | \
* sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
* sudo apt update
* sudo apt install docker-compose-plugin -y
* docker compose version

1. Block Scout

cd ./docker-compose

docker-compose up -d

cd ./docker

make start

cd ./docker

make stop

Mint wallet:

0x651550A72B3cA9247D9012017d6676bC6418103a

Vault wallets:

0xBdBF2e14d9f4Aa400760cb70Ea6c38a483389aef

0x9f1175FA421c14C5DDf4C7198e89c75f80540Fbf

0x5f5C7d84B9f6cDE23c93efc789A33093341c3b17