

wolfecoin Blockchain

In this activity I demonstrate how to set up a testnet blockchain and send a transaction between accounts.

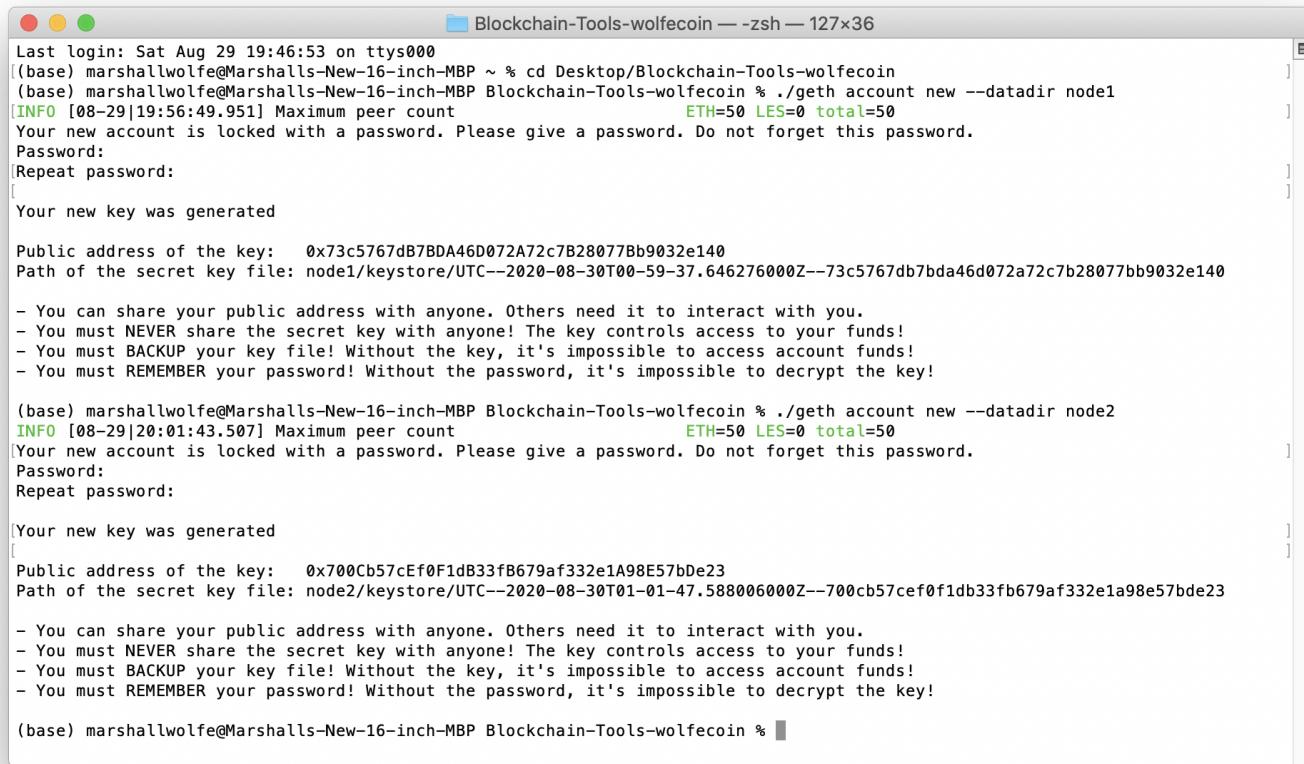
Requirements:

- Blockchain-Tools including **geth** and **puppeth** must be installed. In this exercise I have created a copy called **Blockchain-Tools-wolfecoin** for illustrative purposes.

Step 1: Create accounts for two (or more) nodes for the network with a separate datadir for each using geth

1. Open a terminal window and navigate to Blockchain-Tools-wolfecoin
2. Run the following command: `./geth account new --datadir node1`
3. Enter the password `wolfecoin`
4. Confirm the password `wolfecoin`
5. Take note of the Public address of the key and the Path of the secret key file that are generated for node1
6. Run the following command: `./geth account new --datadir node2`
7. Enter the password `wolfecoin`
8. Confirm the password `wolfecoin`
9. Take note of the Public address of the key and the Path of the secret key file that are generated for node2

Screenshot of geth configuration:



```
Last login: Sat Aug 29 19:46:53 on ttys000
(base) marshallwolfe@Marshalls-New-16-inch-MBP ~ % cd Desktop/Blockchain-Tools-wolfecoin
(base) marshallwolfe@Marshalls-New-16-inch-MBP Blockchain-Tools-wolfecoin % ./geth account new --datadir node1
[INFO] [08-29|19:56:49.951] Maximum peer count ETH=50 LES=0 total=50
Your new account is locked with a password. Please give a password. Do not forget this password.
Password:
Repeat password:
Your new key was generated

Public address of the key: 0x73c5767dB7BDA46D072A72c7B28077Bb9032e140
Path of the secret key file: node1/keystore/UTC--2020-08-30T00-59-37.646276000Z--73c5767db7bda46d072a72c7b28077bb9032e140

- You can share your public address with anyone. Others need it to interact with you.
- You must NEVER share the secret key with anyone! The key controls access to your funds!
- You must BACKUP your key file! Without the key, it's impossible to access account funds!
- You must REMEMBER your password! Without the password, it's impossible to decrypt the key!

(base) marshallwolfe@Marshalls-New-16-inch-MBP Blockchain-Tools-wolfecoin % ./geth account new --datadir node2
[INFO] [08-29|20:01:43.507] Maximum peer count ETH=50 LES=0 total=50
Your new account is locked with a password. Please give a password. Do not forget this password.
Password:
Repeat password:
Your new key was generated

Public address of the key: 0x700Cb57cEf0F1dB33fB679af332e1A98E57bDe23
Path of the secret key file: node2/keystore/UTC--2020-08-30T01-01-47.588006000Z--700cb57cef0f1db33fb679af332e1a98e57bde23

- You can share your public address with anyone. Others need it to interact with you.
- You must NEVER share the secret key with anyone! The key controls access to your funds!
- You must BACKUP your key file! Without the key, it's impossible to access account funds!
- You must REMEMBER your password! Without the password, it's impossible to decrypt the key!

(base) marshallwolfe@Marshalls-New-16-inch-MBP Blockchain-Tools-wolfecoin %
```

Step 2: Use puppeth to generate a genesis block

1. Open a terminal window and navigate to Blockchain-Tools-wolfecoin

2. Run the following command: `./puppeth`
3. Type `2` to pick the Configure new genesis
4. Type `1` to Create new genesis from scratch
5. Type `2` to choose the Clique (Proof of Authority) consensus algorithm
6. Press `Enter` to choose the default =15 seconds blocks should take
7. Paste both account addresses for node1 and node2 one at a time into the list of accounts to seal. Press `Enter` to move to the next step
8. Paste them again in the list of accounts to pre-fund since there are no block rewards in Proof of Authority. Press `Enter` to move to the next step
9. Choose `no` for pre-funding the pre-compiled accounts (0x1 .. 0xff) with wei to keep the genesis cleaner
10. Choose the default = random chain/network ID by pressing `Enter`
11. Type `2` to choose Manage existing genesis
12. Type `2` to Export genesis configurations
13. Press `Enter` to export to the default = current folder
14. You can delete the `wolfecoin-harmony.json` file because we only need the `wolfecoin.json` file

Screenshot of puppeth configuration:

```
Blockchain-Tools-wolfecoin — puppeth — 115x88
Last login: Sat Aug 29 19:35:06 on ttys000
[(base) marshallwolfe@Marshalls-New-16-inch-MBP ~ % cd Desktop/Blockchain-Tools-wolfecoin
[(base) marshallwolfe@Marshalls-New-16-inch-MBP Blockchain-Tools-wolfecoin % ./puppeth
[+-----+
| Welcome to puppeth, your Ethereum private network manager |
|
| This tool lets you create a new Ethereum network down to |
| the genesis block, bootnodes, miners and ethstats servers |
| without the hassle that it would normally entail. |
|
| Puppeth uses SSH to dial in to remote servers, and builds |
| its network components out of Docker containers using the |
| docker-compose toolset. |
+-----+
Please specify a network name to administer (no spaces, hyphens or capital letters please)
> wolfecoin

Sweet, you can set this via --network=wolfecoin next time!

INFO [08-29|19:47:21.273] Administering Ethereum network
WARN [08-29|19:47:21.275] No previous configurations found
name=wolfecoin
path=/Users/marshallwolfe/.puppeth/wolfecoin

What would you like to do? (default = stats)
1. Show network stats
2. Configure new genesis
3. Track new remote server
4. Deploy network components
> 2

What would you like to do? (default = create)
1. Create new genesis from scratch
2. Import already existing genesis
> 1

Which consensus engine to use? (default = clique)
1. Ethash - proof-of-work
2. Clique - proof-of-authority
> 2

How many seconds should blocks take? (default = 15)
>

Which accounts are allowed to seal? (mandatory at least one)
> 0x73c5767dB7BDA46D072A72c7B28077Bb9032e140
> 0x700Cb57cEf0F1dB33fB679af332e1A98E57bDe23
> 0x

Which accounts should be pre-funded? (advisable at least one)
> 0x73c5767dB7BDA46D072A72c7B28077Bb9032e140
> 0x700Cb57cEf0F1dB33fB679af332e1A98E57bDe23
> 0x

Should the precompile-addresses (0x1 .. 0xff) be pre-funded with 1 wei? (advisable yes)
> no

Specify your chain/network ID if you want an explicit one (default = random)
>
INFO [08-29|20:16:13.064] Configured new genesis block

What would you like to do? (default = stats)
1. Show network stats
2. Manage existing genesis
3. Track new remote server
4. Deploy network components
> 2

1. Modify existing configurations
2. Export genesis configurations
3. Remove genesis configuration
> 2

Which folder to save the genesis specs into? (default = current)
Will create wolfecoin.json, wolfecoin-aleth.json, wolfecoin-harmony.json, wolfecoin-parity.json
>
INFO [08-29|20:17:38.313] Saved native genesis chain spec
path=wolfecoin.json
ERROR[08-29|20:17:38.313] Failed to create Aleth chain spec
err="unsupported consensus engine"
ERROR[08-29|20:17:38.313] Failed to create Parity chain spec
err="unsupported consensus engine"
INFO [08-29|20:17:38.313] Saved genesis chain spec
client=harmony path=wolfecoin-harmony.json

What would you like to do? (default = stats)
1. Show network stats
2. Manage existing genesis
3. Track new remote server
4. Deploy network components
> █
```

Step 3: Initialize each node with the new wolfecoin.json with geth

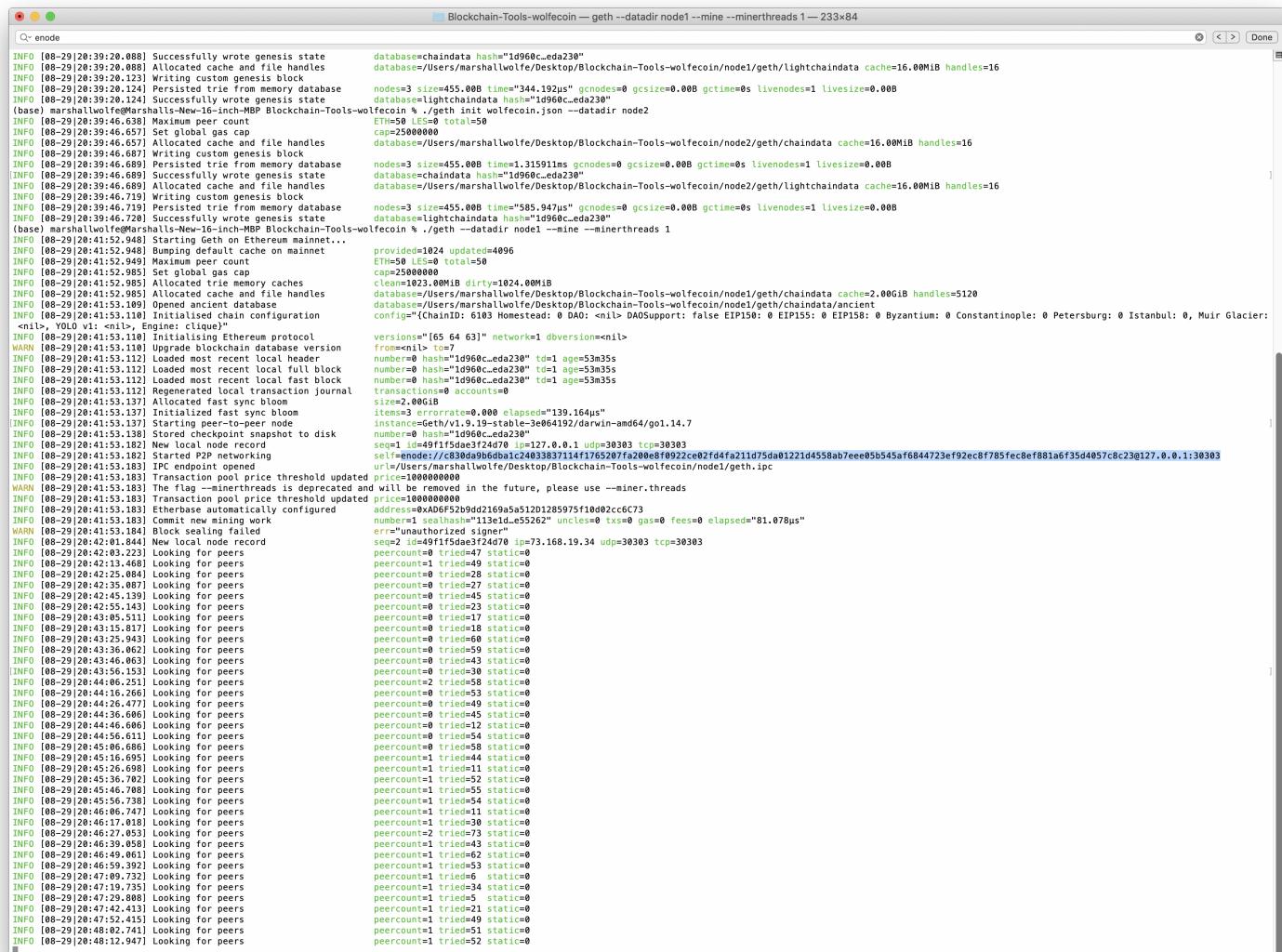
1. Open a terminal window and navigate to Blockchain-Tools-wolfecoin
2. At this point, I determined that I needed to re-create node1 and node2 so I went back to repeat Step 1, and then proceeded to...
3. Run the following command: `./geth init wolfecoin.json --datadir node1`
4. Run the following command to initialize node1: `./geth init wolfecoin.json --datadir node1`
5. Run the following command to initialize node2: `./geth --datadir node1 --mine --minerthreads 1`
6. Run the following command to unlock the account and enable mining on the first node: `./geth --datadir node1 --mine --minerthreads 1`
7. Copy the enode that is generated
8. Open another terminal window and navigate to Blockchain-Tools-wolfecoin

9am . Run the following command to unlock the account and enable mining on the second node:

```
./geth --datadir node2 --port 30304 --rpc --bootnodes "enode://c830da9b6dba1c24033837114f1765207fa200e8f0922ce02fd4fa211d75da01221c
```

9. You should now see that both nodes are producing new blocks!

Screenshot of node1:



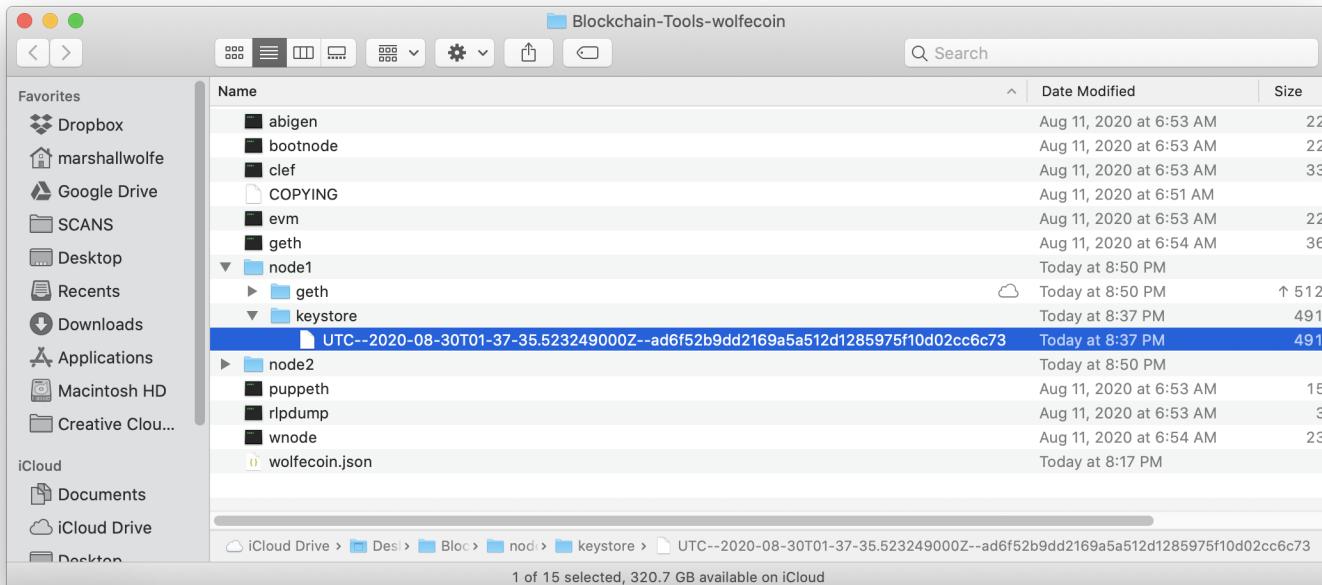
```
INFO [08-29|20:39:20.088] Successfully wrote genesis state
INFO [08-29|20:39:20.120] Allocating cache and file handles
INFO [08-29|20:39:20.123] Writing initial genesis block
INFO [08-29|20:39:20.124] Persisted trie from memory database
INFO [08-29|20:39:20.124] Successfully wrote genesis state
(base) marshallwolfe@Marshall-NEW-16-inch-MBP:~/Blockchain-Tools-wolfecoin$ ./geth init wolfecoin.json --datadir node2
INFO [08-29|20:39:46.639] MaxPeer count
INFO [08-29|20:39:46.639] Set global gas cap
INFO [08-29|20:39:46.657] Allocated cache and file handles
INFO [08-29|20:39:46.657] Writing custom genesis block
INFO [08-29|20:39:46.689] Persisted trie from memory database
INFO [08-29|20:39:46.689] Successfully wrote genesis state
INFO [08-29|20:39:46.689] Allocate cache and file handles
INFO [08-29|20:39:46.700] Set global gas cap
INFO [08-29|20:39:46.720] Persisted trie from memory database
INFO [08-29|20:39:46.720] Successfully wrote genesis state
(base) marshallwolfe@Marshall-NEW-16-inch-MBP:~/Blockchain-Tools-wolfecoin$ ./geth init wolfecoin.json --datadir node1
INFO [08-29|20:41:52.948] Starting Geth on Ethereum mainnet...
INFO [08-29|20:41:52.948] Bumping default cache on mainnet
INFO [08-29|20:41:52.948] MaxPeer count
INFO [08-29|20:41:52.985] Set global gas cap
INFO [08-29|20:41:52.985] Allocated tri memory caches
INFO [08-29|20:41:52.985] Allocated cache and file handles
INFO [08-29|20:41:53.100] Opened ancient database
INFO [08-29|20:41:53.100] Initiating chain configuration
<n1>:YOL0.v1n1: Enode: clique: 0
INFO [08-29|20:41:53.137] Initializing Ethereum protocol
WARN [08-29|20:41:53.138] Upgrade blockchain database version
INFO [08-29|20:41:53.138] Starting local keystore snapshot to disk
INFO [08-29|20:41:53.138] Local keystore snapshot to disk
INFO [08-29|20:41:53.138] Loaded most recent local full block
INFO [08-29|20:41:53.138] Regenerated local fast block
INFO [08-29|20:41:53.138] Regenerated local transaction journal
INFO [08-29|20:41:53.137] Allocated fast sync bloom
INFO [08-29|20:41:53.137] Initialized fast sync bloom
INFO [08-29|20:41:53.137] Starting peer-to-peer node
INFO [08-29|20:41:53.137] Starting local P2P endpoint snapshot to disk
INFO [08-29|20:41:53.138] Local P2P endpoint snapshot to disk
INFO [08-29|20:41:53.182] Started P2P networking
INFO [08-29|20:41:53.183] IPC endpoint opened
INFO [08-29|20:41:53.183] Transaction pool price threshold updated
INFO [08-29|20:41:53.183] The flag --minerthreads is deprecated and will be removed in the future, please use --miner.threads
INFO [08-29|20:41:53.183] Transaction pool price threshold updated
INFO [08-29|20:41:53.183] Transaction pool price threshold updated
INFO [08-29|20:41:53.183] Block sealing automatically configured
INFO [08-29|20:41:53.183] Block sealing failed
INFO [08-29|20:42:01.844] New local node record
INFO [08-29|20:42:01.844] Looking for peers
INFO [08-29|20:43:13.081] Looking for peers
INFO [08-29|20:42:25.884] Looking for peers
INFO [08-29|20:42:35.087] Looking for peers
INFO [08-29|20:42:45.139] Looking for peers
INFO [08-29|20:42:55.143] Looking for peers
INFO [08-29|20:43:05.511] Looking for peers
INFO [08-29|20:43:17.077] Looking for peers
INFO [08-29|20:43:25.943] Looking for peers
INFO [08-29|20:43:36.062] Looking for peers
INFO [08-29|20:43:46.063] Looking for peers
INFO [08-29|20:43:56.153] Looking for peers
INFO [08-29|20:44:16.266] Looking for peers
INFO [08-29|20:44:26.477] Looking for peers
INFO [08-29|20:44:36.606] Looking for peers
INFO [08-29|20:44:46.606] Looking for peers
INFO [08-29|20:44:56.611] Looking for peers
INFO [08-29|20:45:16.695] Looking for peers
INFO [08-29|20:45:26.698] Looking for peers
INFO [08-29|20:45:36.702] Looking for peers
INFO [08-29|20:45:46.708] Looking for peers
INFO [08-29|20:45:56.730] Looking for peers
INFO [08-29|20:46:17.871] Looking for peers
INFO [08-29|20:46:17.818] Looking for peers
INFO [08-29|20:46:27.053] Looking for peers
INFO [08-29|20:46:39.058] Looking for peers
INFO [08-29|20:46:49.061] Looking for peers
INFO [08-29|20:46:59.062] Looking for peers
INFO [08-29|20:47:09.722] Looking for peers
INFO [08-29|20:47:19.735] Looking for peers
INFO [08-29|20:47:29.888] Looking for peers
INFO [08-29|20:47:42.413] Looking for peers
INFO [08-29|20:47:52.415] Looking for peers
INFO [08-29|20:48:02.741] Looking for peers
INFO [08-29|20:48:12.947] Looking for peers
```

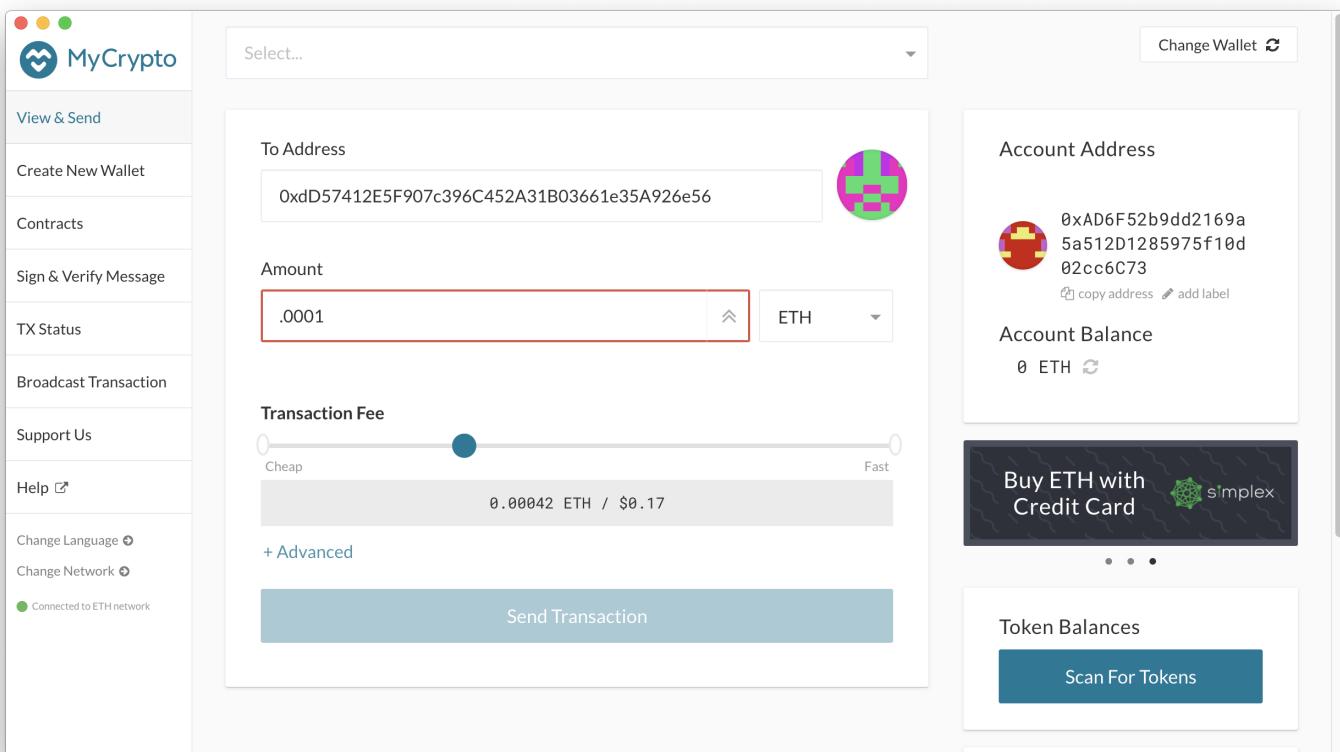
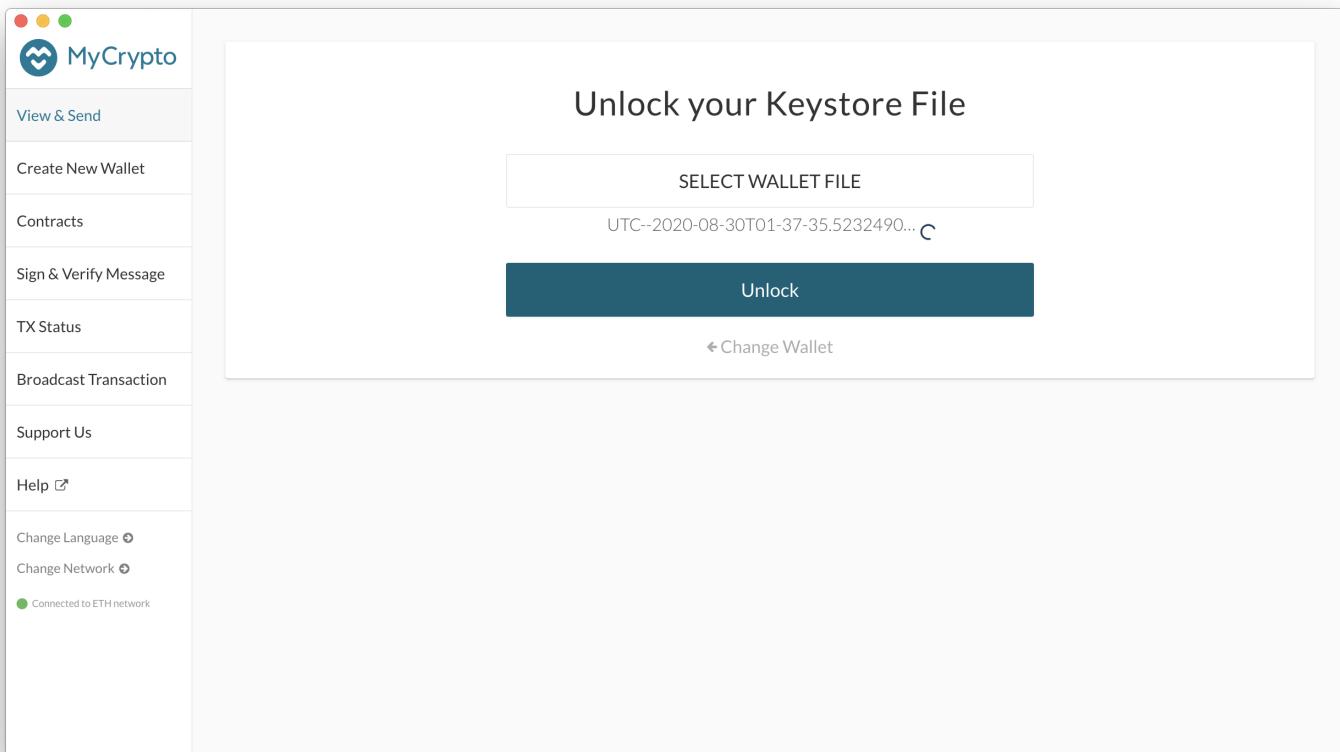
Screenshot of node2:

Step 4: Send a test transaction

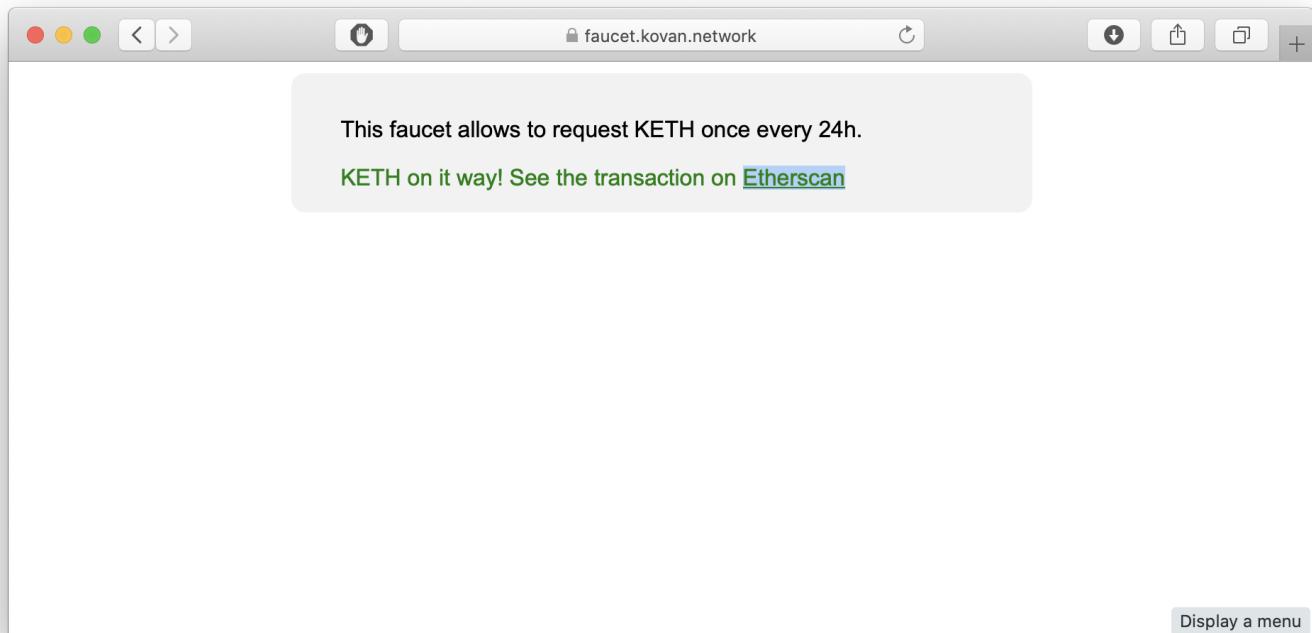
1. Use the MyCrypto GUI wallet to connect to the node1 by importing the keystore file from the node1/keystore directory directly into MyCrypto which will import the private key
 2. Enter the password `wolfecoin`

Screenshot of node1/keystore:

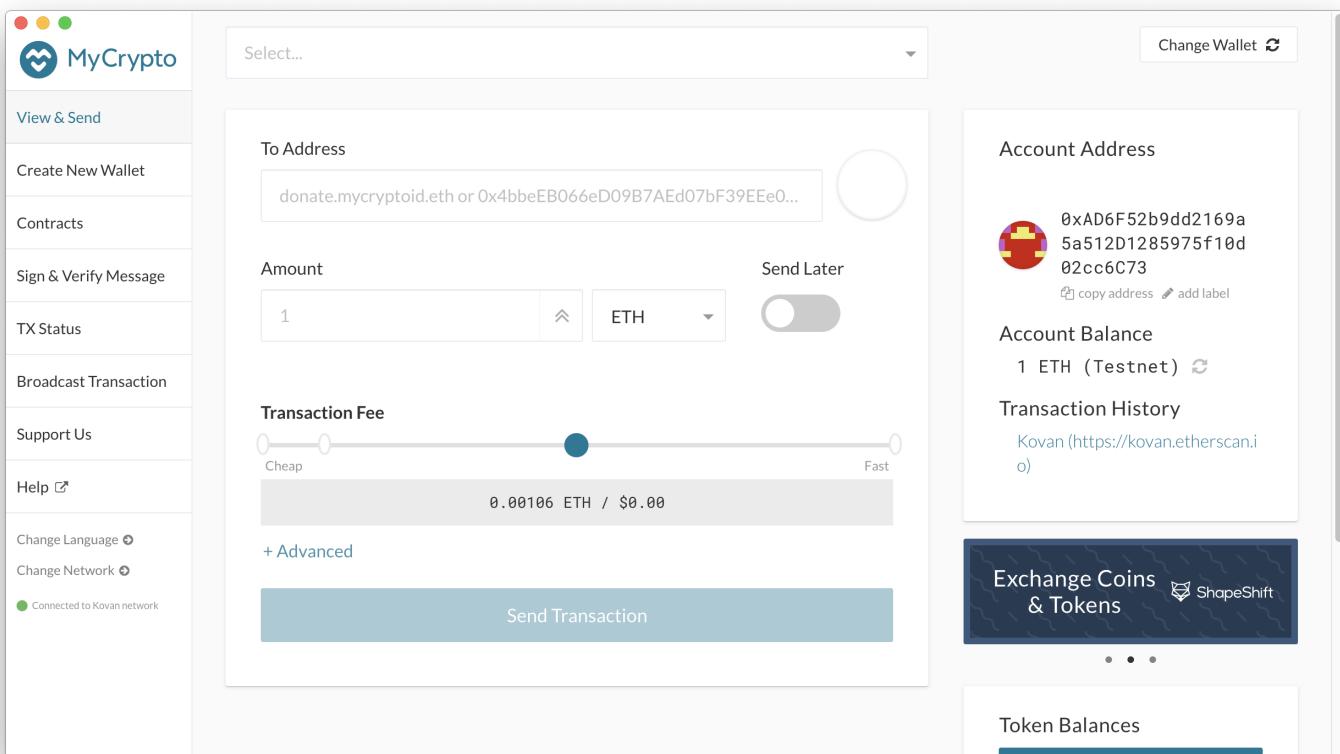




3. After noticing that the Account Balance was 0 ETH, I used <https://faucet.kovan.network/> to fund the account with ETH



4. Changing to the Kovan network, I could now see 1 ETH in the Account Balance:



MyCrypto

Select...

Change Wallet ↗

To Address

donate.mycryptoid.eth or 0x4bbeEB066eD09B7AE07bF39EEe0...

Amount

1 ETH Send Later

Transaction Fee

Cheap Fast

0.00106 ETH / \$0.00

+ Advanced

Send Transaction

Account Address

0xAD6F52b9dd2169a 5a512D1285975f10d 02cc6C73

copy address add label

Account Balance

1 ETH (Testnet) ↗

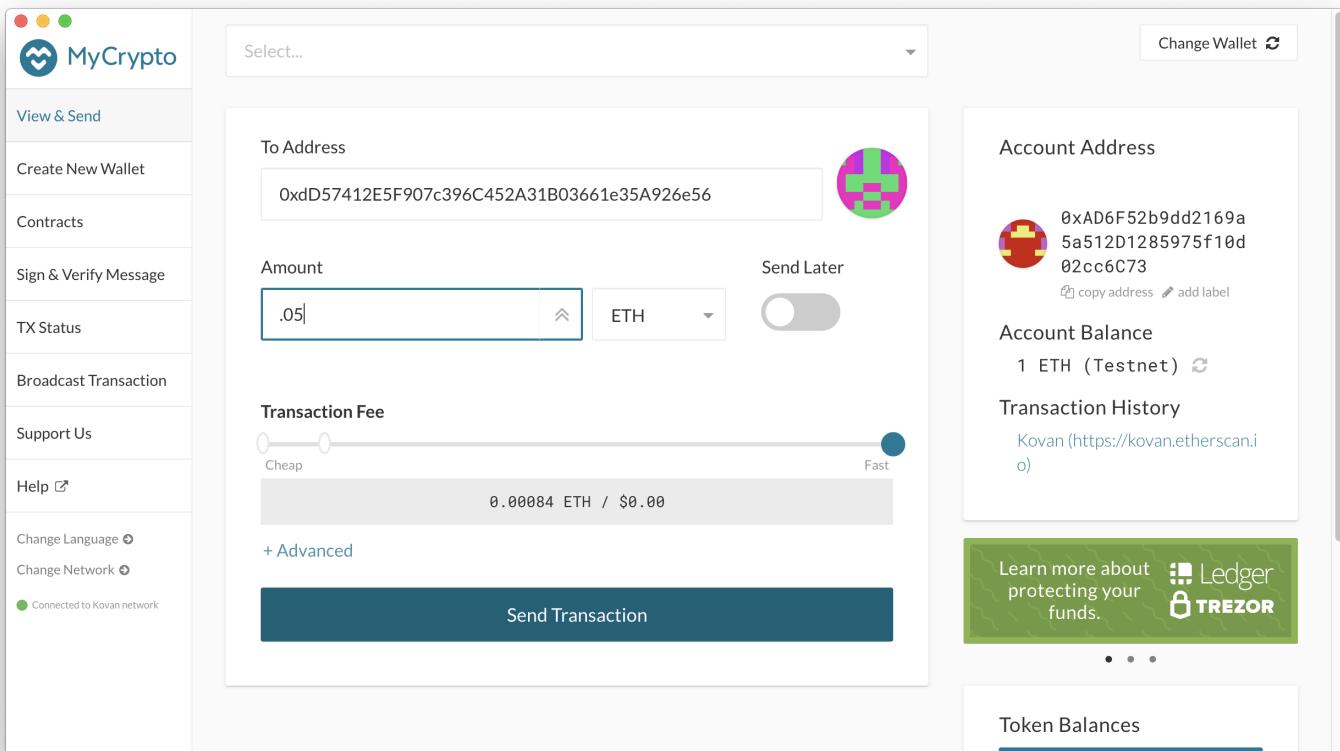
Transaction History

Kovan (<https://kovan.etherscan.io>)

Exchange Coins & Tokens ShapeShift

Token Balances

5. Send a transaction from the node1 account to the node2 account by pasting the Public address of the key for node2 into the To Address field, entering an amount of **.05** , and increasing the Transaction Fee to **0.00084** ETH for a speedy transaction



MyCrypto

Select...

Change Wallet ↗

To Address

0xd57412E5F907c396C452A31B03661e35A926e56

Amount

.05 ETH Send Later

Transaction Fee

Cheap Fast

0.00084 ETH / \$0.00

+ Advanced

Send Transaction

Account Address

0xAD6F52b9dd2169a 5a512D1285975f10d 02cc6C73

copy address add label

Account Balance

1 ETH (Testnet) ↗

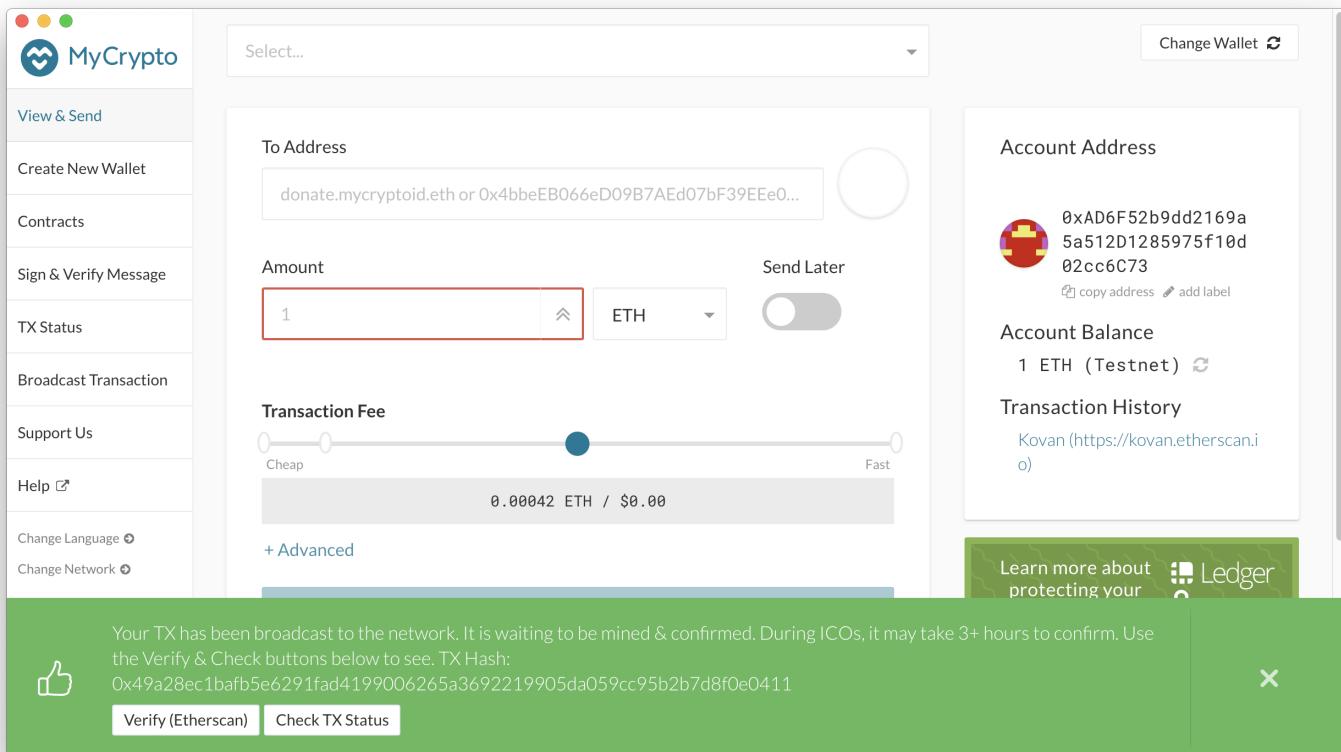
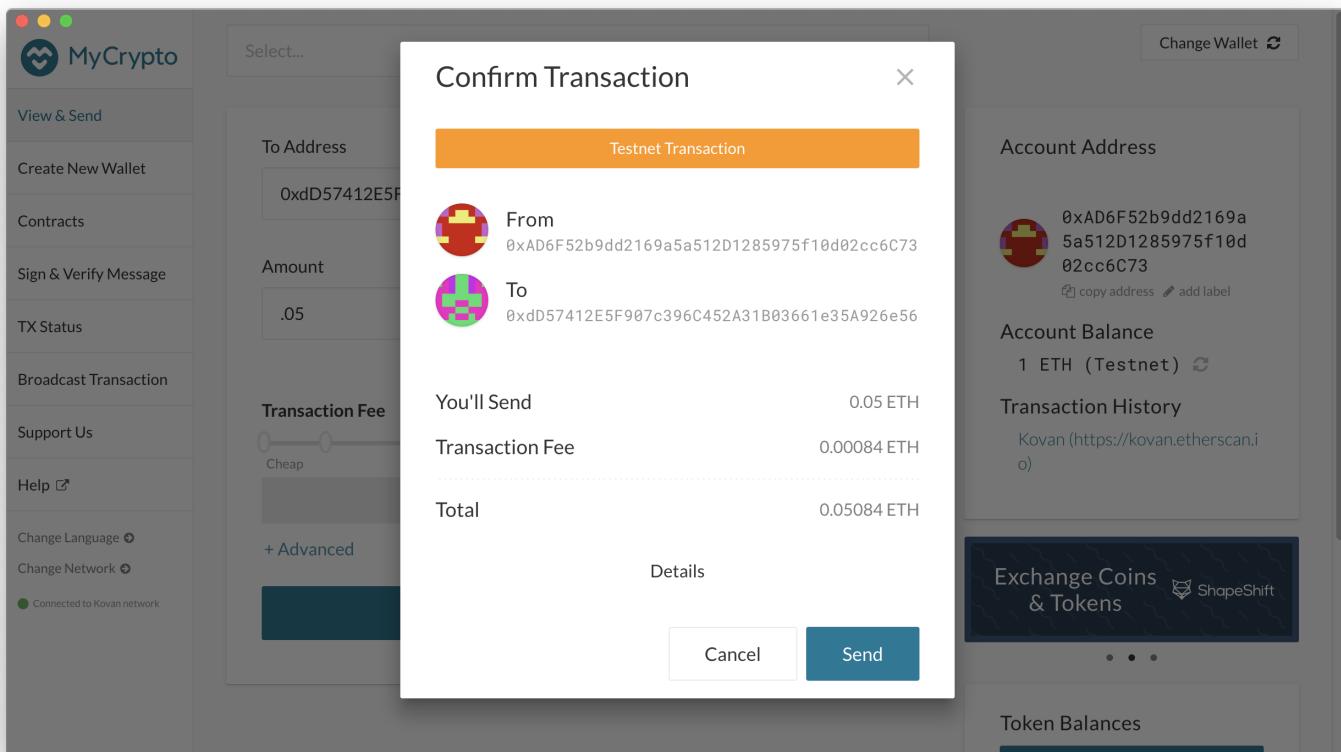
Transaction History

Kovan (<https://kovan.etherscan.io>)

Learn more about protecting your funds. 

Token Balances

6. Choose **Send Transaction** and confirm by choosing **Send**



7. Copy the transaction hash and paste it into the "TX Status" section of the app, or click "TX Status" in the popup

MyCrypto

View & Send

Create New Wallet

Contracts

Sign & Verify Message

TX Status

Broadcast Transaction

Support Us

Help

Change Language ⓘ

Change Network ⓘ

Connected to Kovan network

Check Transaction Status

Enter your Transaction Hash to check on its status. If you don't know your Transaction Hash, you can look it up on [Etherscan](#) by looking up your address.

Aug 29, 2020 9:12 PM - 0xad6f52... to 0xd5741... x ▾

or

0x49a28ec1bafb5e6291fad4199006265a3692219905da059cc95b2b7d8f0e0411

Check TX Status

Status	SUCCESSFUL
TX Hash	0x49a28ec1bafb5e6291fad4199006265a3692219905da059cc95b2b7d8f0e0411
Block Number	20578821
From Address	 0xAD6F52b9dd2169a5a512D1285975f10d02cc6C73
To Address	 0xd57412E5F907c396C452A31B03661e35A926e56
Amount	0.05 ETH
Gas Price	40 Gwei
Gas Limit	21000
Gas Used	21000
Transaction Fee	0.00084 ETH

The transaction was successful!