DEMO (destination: NameReg service)

Simple app 🡪 Controller app 🡪 Destination

User on local machine will use simple browser application to talk to Blockchain applications

Environments (4) and their corresponding components:

**LOCAL**

1. Simple HTML application (to become wallet)

**ERIS**

1. Requester application (Eris Verify Oraclizer)R
2. NameReg application (identity.js)
3. Eris keys service
4. ErisBigchainService

**TOMCAT**

1. Controller application (proxy)

**BIGCHAINDB**

1. BigChainDB driver

\*NOTE: This cannot handle multiple requests at the same time. After your first request, make sure to wait a little before a future request!

**Preconditions**: Make sure the appropriate chain is on. Currently, we use the chain: **newchain2**.

In order to check identity.js is working as expected:

1. Note, if (and only if) you see a rethinkdb error in BigchainDB, run this command in bighcainDB.

* CMD : **nohup rethinkdb --bind 10.100.98.217 --driver-port 28015 --cluster-port 29015 --http-port 8081 &**

1. Run the RequesterApp.js:

* /home/demoadmin/ .eris/apps/SOLIDITY/VerifyOraclizerEthereum/wallet2
* CMD: **node RequesterApp.js**

1. Run identity.js (NameReg):

* /home/demoadmin/.eris/apps/nameRegDEMO
* CMD: **node identity.js**

1. Run BigchainDriver.py (In the Bigchain Server: 10.101.114.230):

* /home/demoadmin
* CMD: **python BigchainDriver.py**

1. Run BigchainEris

* /home/demoadmin/.eris/apps/SOLIDITY/BigchainOraclizer
* CMD: **node ErisBigchainService.js**

1. Run the Proxy (In the Tomcat Server: 10.101.114.231):

* /home/demoadmin/To Cincinnati/proxy
* Note, in demoadmin, type: cd To\ Cincinnatti/
* CMD: **node index.js**

1. Open up C:\Users\1113219\Desktop\To Cincinnatti\simpleApp\index.html in Chrome, make sure CORS is enabled.

**Example Input**

**Transaction Type:** NamespaceRegister

**Address of Name Registry Contract:** D29FE32C8B8D99D96D37C464C6491287A25AC10E

**Desired Name:** Blockchains4Life

**msg** = "8836a77b68579d1d8d4427c0cda24960f6c123f17ccf751328cc621d6237da22";

**pubKey** = "02d7ceb37a16fde15a5237652b31a52d94def283a2ab09aaf5d1af48db1b84e20a";

**sig** = "7051442bbf18bb2c86cbc8951a07e27ec6ba05ac3fa427e4c6b948e3dcf91a94046b048edf52445fb22cc776a94b87c3f55426f993458ec744f61f09fb46eeaa";

If it is desired via curl:

curl --request PUT --data '{"pub":"61B916299F8978D498424CC8CEFA4B5CB33EB920959524144BCAEB7F4AC75E5D","sig": "A3C7613BB4EEAA948EE18436058A4EC66BA4DD20FA7A9D825CB09165516CCA4F7C3337F05723F497EE199EF43C5CC18F607115EF7AA58965CFF24E9EE59DDA04", "msg": "E3B0C44298FC1C149AFBF4C8996FB92427AE41E4649B934CA495991B7852B855","input":"NamespaceRegister","address":"D29FE32C8B8D99D96D37C464C6491287A25AC10E","name":"spencer"}' 10.100.99.207:8003/toRegister