**Handover Document**

**10/6/2016**

Make sure to have a Webex with Arun in the morning. He will get you guys up to speed on how to use the wallet if you have any blockers. In particular, remember you must run the command, **npm run dev**, to use your wallet.

Note: The DAO is fully deployed, with correct validators in them.

To test the wallet, you must do the following:

1. Make sure you are in Tomcat

You must also do all curl requests to Tomcat.

**VPN IP**: 10.100.98.208

**Baseline IP**: 10.101.114.231

**Turn on the chain:** eris chains start newchain4

1. In the (Tomcat) directory:

/home/demoadmin /Migration\_from\_Eris/VerifyOraclizerEthereum/wallet2

Run the command: **node RequesterApp.js**

This runs the Verification Oraclizer.

1. In the (Tomcat) directory:

/home/demoadmin/Migration\_from\_Eris/BigchainOraclizer

Run the command: **node ErisBigchainService.js**

This runs the Bigchain Oraclizer.

1. In the Bigchain Environment

(**VPN IP: 10.100.98.217**

**Baseline IP: 10.101.114.230**)

In the main directory (the one when you join):

/home/demoadmin

Run the command: **python BigchainDriver.py**

This runs the Bigchain Driver.

1. In the (Tomcat) directory:

/home/demoadmin/Migration\_from\_Eris/gatekeeper

Run the command: **node gatekeeper\_v2.js**

Also, run the command: **node ballot.js**

1. In the (Tomcat) directory:

/home/demoadmin/DigitalTwin

Run the command: **node index.js**

This is for the digital twin routing.

Now, please keep the following in mind while you debug:

\*Make sure your Unique ID is different every time. You can’t submit the same Unique ID twice!

\*In gatekeeper\_v2.js, you will see there is a hardcoded formdata1 object at line 658. This is for debugging purposes. If you don’t want to send the request from the wallet, you can change the object name from formdata1 to formdata, and comment out the line at 656 (var formdata = req.body).

\*If you need to redeploy a contract, you can do with the following command:

eris pkgs do --chain newchain4 --address BB6724FE1A9A2A6D4F95B9969EAC59FDA94838AD --compiler <https://compilers.monax.io:10114>

\*If you redeploy the DAO, you will have to put in the appropriate validators (see my document in this slack channel: *TOMCATErisEnvironment.docx*)

\*You can see my previous documents shared in the past week on how to test ballot voting, but remember to change the IP address as we are now in tomcat.

Here is a reminder of the accounts to use:

Testing

We use these three accounts. They have been added to the Dao.

**Account 1**

String of public key is: 0373ecbb94edf2f4f6c09f617725e7e2d2b12b3bccccfe9674c527c83f50c89055

String of private key is: 66cdd508f950d08e02d8448c55a03c14e08e9a3447c4361105e14ad32ec3286a

Message hash string: 7624778dedc75f8b322b9fa1632a610d40b85e106c7d9bf0e743a9ce291b9c6f

sig string is: c21bb2c81d0ca548a59b61a6d9fdc3871b867a951963e059a1862f829e4f835a5a0584405f64288ca6315d871efc4d7c39a625c5eb01108f2ddb49e37a711515

**Account 2**

String of public key is: 03683536757fdb821c10810b51caa51a84fc1dfab5c17edbf5246f9713ffe31adf

String of private key is: f4822c62077945009cd816e479eef4ac371613d2d44164fc7ecaa93f180f2eb4

Message hash string: 7624778dedc75f8b322b9fa1632a610d40b85e106c7d9bf0e743a9ce291b9c6f

sig string is: 9a83145f3533169aea22d253ee30533768f112b0818d096716f7d6ef664d95ea4c8e2e548c57d0142b82ce85403d307cb817a9d9e7b64e088d8c4b08d551eab4

**Account 3**

String of public key is: 03a066efbb37f5fabfab05bf4a65e0dc376d0e3fb1c3d930d7f5ec6da3ac5bc237

String of private key is: 2eff28029dced258c4aca03415fd3ef201ec118e57241b758e82454a6bc75cd4

Message hash string: 7624778dedc75f8b322b9fa1632a610d40b85e106c7d9bf0e743a9ce291b9c6f

sig string is: 9337df4f95d3460dea440315c17666ebef89db10d19760ee3277c4fd1fafed7d1e3b1a8b7822bb8374de88b3b50305d7d94d14377b8b0945f1180a8db45bd0b4