**MyCOID.js Testing**

**TODO: Add signature verification.**

Core Identity Setup

For setting up the Core Identity, refer to the file,

*CoreIdentity.js*

This is the address of our Core Identity Contract:

BD50CD907AD2C5A472EAC538B6FE16106794C390

We use the following identities:

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Value** | **Hashed Value** |
| A1 | “a1” | 37D3424576BAFB5FD5F9F8E99478F66780477FCD8D71CB2319B37A64A01640DB |
| A2 | “a2” | A2060FAA0FC5697BC282C626D908A989DC0D2B79270A5CDC58FBC0AB74C35FAF |
| A3 | “a3” | 4C0096A9174723913927E4420E8A38CA630A7C75320866ADB036C401F44761CC |
| C1 | “c1” | 7C1B16802B304761ACD81E033C1DB5A9721DA4B32959E4A8361983CFACACBBEC |

Since the contracts only deal with the hashed values, it does not matter that the Values are not 66 characters are public keys are from our wallet.

-We initialize the following as **owners**: A1, A2, and A3. Each have stake 3.

-We initialize the following as **controllers**: A1, A2, and C1. Each have 5 control tokens. Note that, since A3 is an owner, the Core Identity contract will make A3 a controller with zero tokens.

Endpoint Testing

We use the file,

MyCOID\_dev.js

Endpoint: /getControllers

**Input:**

curl -X POST -d '{"address":"BD50CD907AD2C5A472EAC538B6FE16106794C390","msg":"asdf","sig":"asdf”}’ http://10.100.98.218/getControllers --header "Content-Type:application/json"

**Output:**

Note, by default, status is null if there is no error. The output is as expected:



Endpoint: /delegate

**Input:** We would like to have the controller “a1” delegate 1 controller to the identity “d1”. We do so with the following request:

curl -X POST -d '{"address":"BD50CD907AD2C5A472EAC538B6FE16106794C390","msg":"asdf","sig":"asdf","controller":"c1","delegatee":"d1","amount":1}' http://10.100.98.218:3012/delegate --header "Content-Type:application/json"

**Output:**

The output is true, which builds confidence. As we will see, we will add confidence to this delegation by the next two tests.

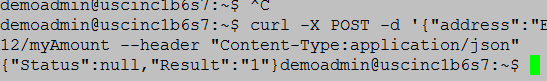


Endpoint: /myAmount

**Input:** This function gives a delegatee the amount of tokens they are delegated for a specific core identity. Based on the previous endpoint call, we expect to see that “d1” has 1 token. The request:

curl -X POST -d '{"address":"BD50CD907AD2C5A472EAC538B6FE16106794C390","msg":"asdf","sig":"asdf","delegatee":"d1"}' http://10.100.98.218:3012/myAmount --header "Content-Type:application/json"

**Output:** As desired, we see that the delegatee “d1” has 1 token.



Endpoint: /amountDelegated

**Input:** This endpoint lets a controller know how many tokens they have delegated. When this is called, we have had controller “c1” delegate two tokens to “d1” (called the delegate endpoint twice). The request is:

curl -X POST -d '{"address":"BD50CD907AD2C5A472EAC538B6FE16106794C390","msg":"asdf","sig":"asdf","controller":"c1"}' http://10.100.98.218:3012/amountDelegated --header "Content-Type:application/json"

**Output:** As desired, we see “c1” has delegated two tokens:



Endpoint: /revokeControlDelegation

**Input:** First, we verify that “d1” has 2 tokens (from “c1”). See below:



Subsequently, we have “c1” revoke one token from “d1”. See the request below:

curl -X POST -d '{"address":"BD50CD907AD2C5A472EAC538B6FE16106794C390","msg":"asdf","sig":"asdf","controller":"c1", "delegatee":"d1","amount":1}' http://10.100.98.218:3012/revokeControlDelegation --header "Content-Type:application/json"

**Output:** Successfully, a token has been revoked.



Endpoint: /myTokenAmount

**Input:** This function tells an owner how many ownership tokens they have. We know that “a1” has three tokens. We make the following request:

curl -X POST -d '{"address":"BD50CD907AD2C5A472EAC538B6FE16106794C390","msg":"asdf","sig":"asdf","owner":"a1"}' http://10.100.98.218:3012/myTokenAmount --header "Content-Type:application/json"

**Output:** As desired, the result is 3.

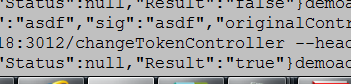


Endpoint: /changeTokenController

**Input:** Recall that this function allows a controller to give tokens to another controller (for that controller to delegate). We have controller “a2” give controller “c1” one token. We make the request:

curl -X POST -d '{"address":"BD50CD907AD2C5A472EAC538B6FE16106794C390","msg":"asdf","sig":"asdf","originalController":"a2","newController":"c1","amount":1}' http://10.100.98.218:3012/changeTokenController --header "Content-Type:application/json"

**Output:** We have a success:

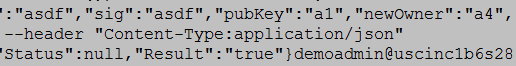


Endpoint: /addOwner

**Input:** We have “a1” add a new owner – “a4” (who is given stake: 1). The request:

curl -X POST -d '{"address":"BD50CD907AD2C5A472EAC538B6FE16106794C390","msg":"asdf","sig":"asdf","pubKey":"a1","newOwner":"a4", "amount":1}' http://10.100.98.218:3012/addOwner --header "Content-Type:application/json"

**Output:** The addition is a success:

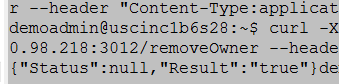


Endpoint: /removeOwner

**Input:** We now have “a1” change their mind and remove “a4”. The request:

curl -X POST -d '{"address":"BD50CD907AD2C5A472EAC538B6FE16106794C390","msg":"asdf","sig":"asdf","pubKey":"a1","owner":"a4"}' http://10.100.98.218:3012/removeOwner --header "Content-Type:application/json"

**Output:** We have a success:

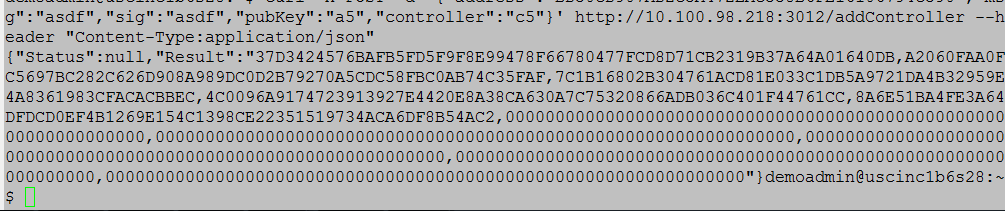


Endpoint: /addController

**Input:** We now have owner “a1” add a controller – “c5”. The request is:

curl -X POST -d '{"address":"BD50CD907AD2C5A472EAC538B6FE16106794C390","msg":"asdf","sig":"asdf","pubKey":"a5","controller":"c5"}' http://10.100.98.218:3012/addController --header "Content-Type:application/json"

**Output:** For debugging, the output is the list of controllers. We see that there is a fourth value, which should be the hash of “c5”.



Endpoint: /removeController

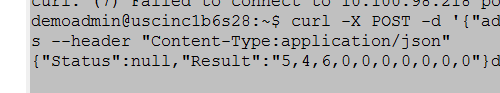
**Input:** We now have owner “a1” remove controller “c2”. The request is:

curl -X POST -d '{"address":"BD50CD907AD2C5A472EAC538B6FE16106794C390","msg":"asdf","sig":"asdf","pubKey"10.100.98.218:3012/removeController --header "Content-Type:application/json"

**Output**: The output is a success:



Calling the getControllers endpoint gives us:



As desired, there are still 15 coins for controllers.

Endpoint: /spendMyTokens

**Input**: We now have delegatee “d1” spend their one remaining token. Note at the time of testing this, the contract was redeployed. Thus, a new COID address was used as shown in the request. The same owners and controllers were instantiated in the core identity, and “d1” was delegated one token by “a1”.

curl -X POST -d '{"address":"B9BC8F4A36739586795CE37262002A6FAB97CF21","msg":"asdf","sig":"asdf","amount":1,"delegatee":"d1"}' http://10.100.98.218:3012/spendMyTokens --header "Content-Type:application/json"

**Output:** As desired:

