崔伟

13601300572@163.com

本文是为标准共识（SNC），就IOST项目所作的技术评测

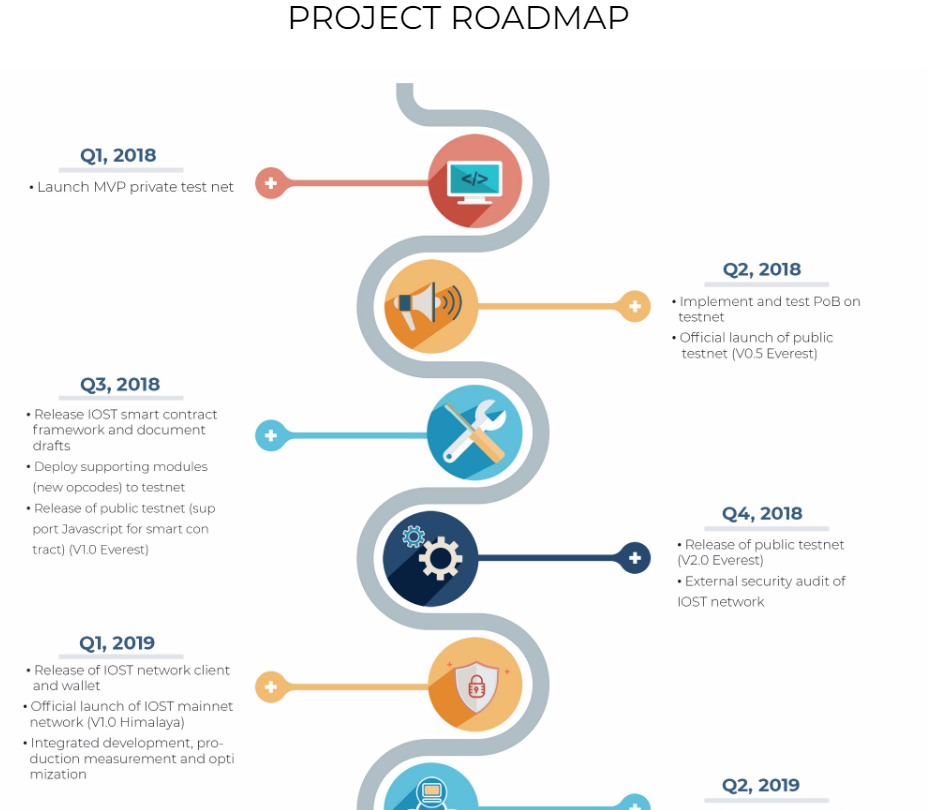
IOST技术评测



# 功能评测

SNC于2018年5月23日发布了对IOST的评测，评级为C。四个多月以来，项目又有了一些新的进展。

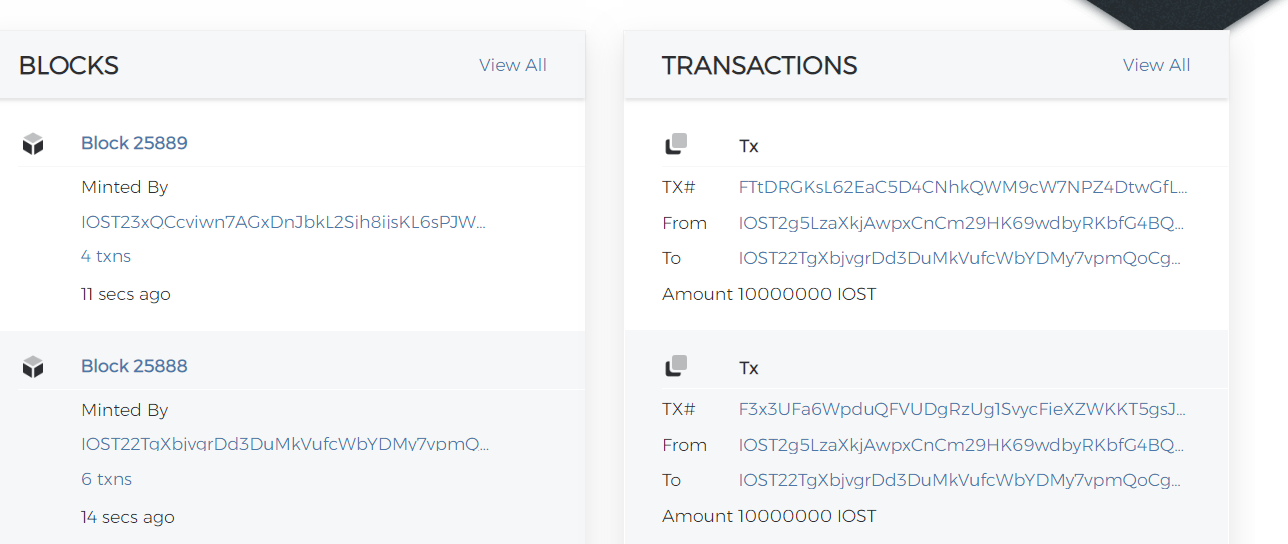
按照其最新路线图：



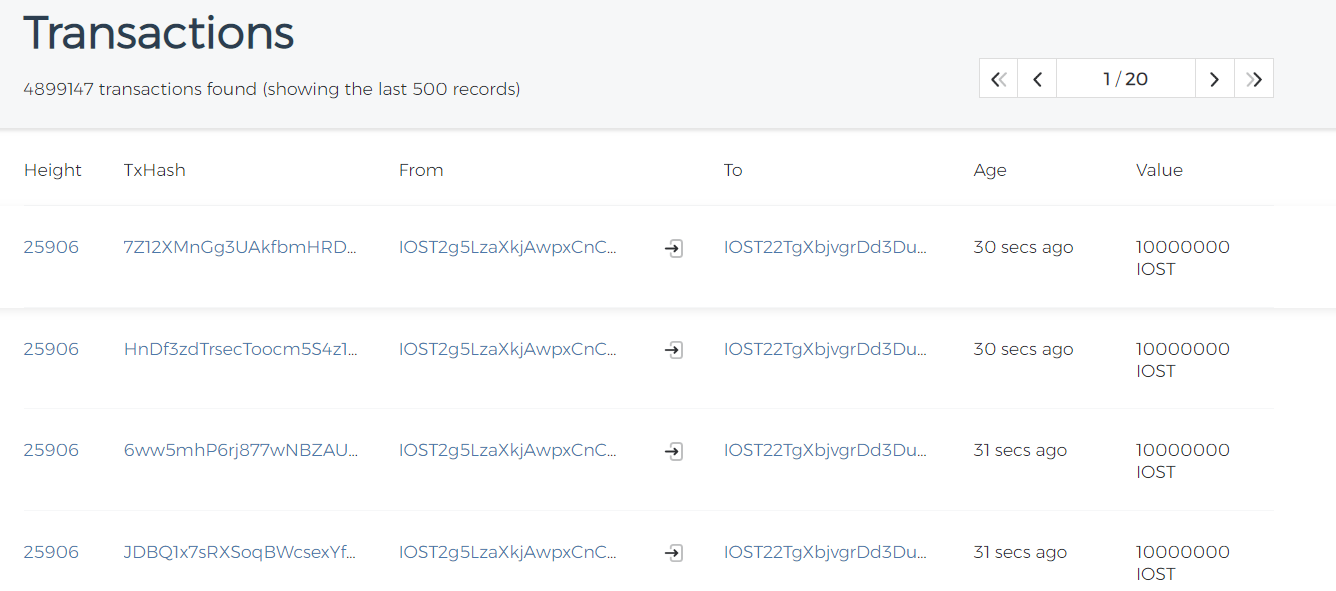
团队在Q1、Q2和Q3的主要任务是：

* 启动测试网络（V0.5 Everest）
* 发布智能合约框架和文档
* 为测试网络部署支持模块
* 发布公共测试网络（支持用JavaScript开发智能合约）

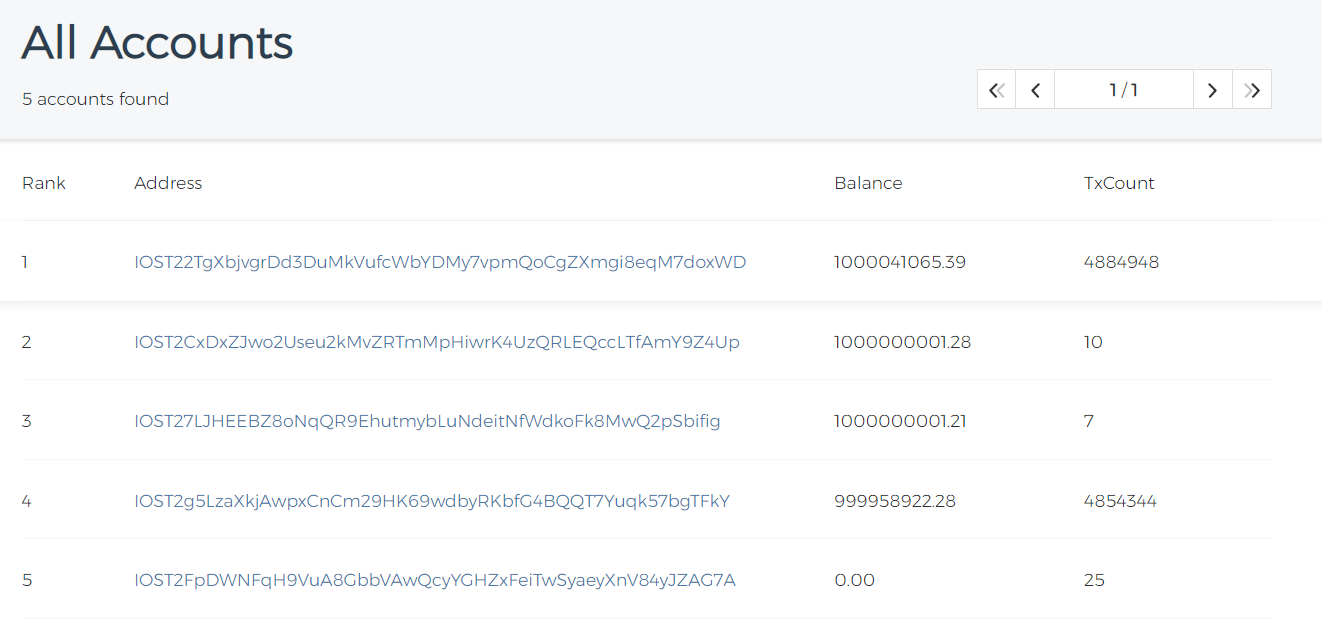
目前，可以从主页的浏览器上看到测试网络运行情况: <https://explorer.iost.io/>。



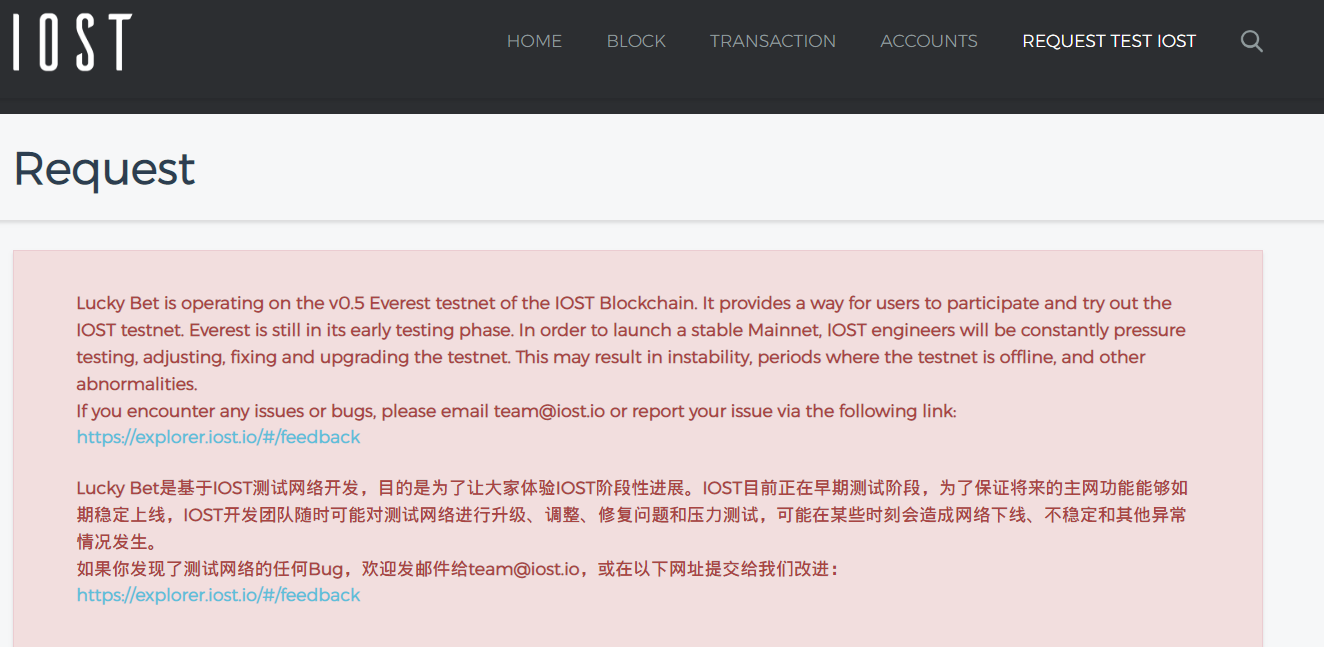
存在交易和挖矿行为，不过从其交易数额来看，全部都是自动执行的交易（金额全部为1000万个IOST）。



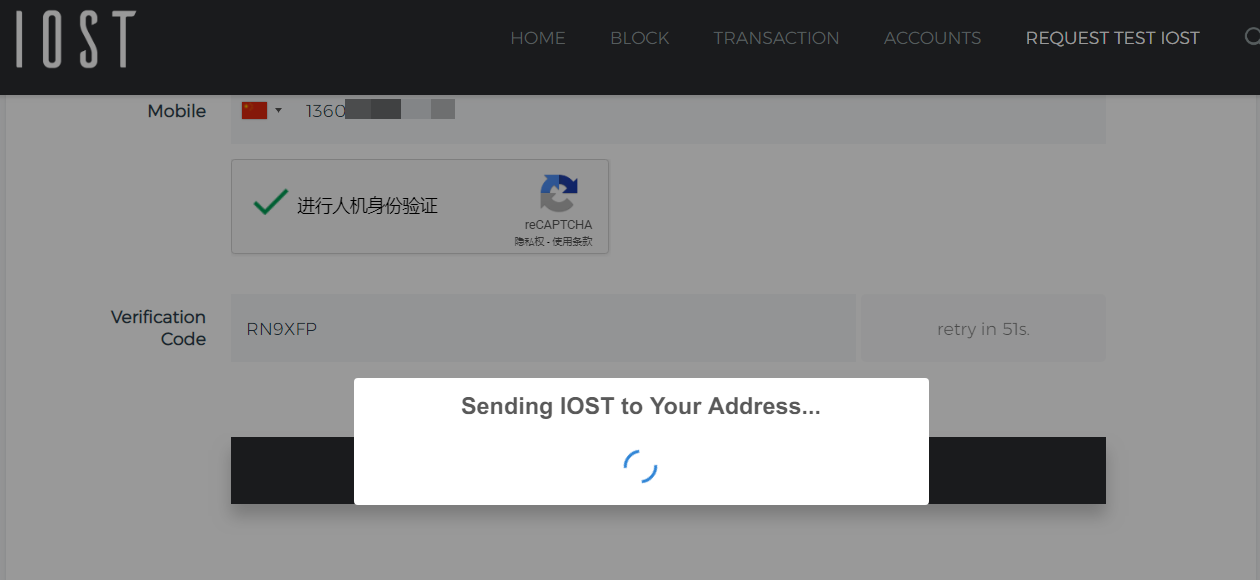
账号只有五个：



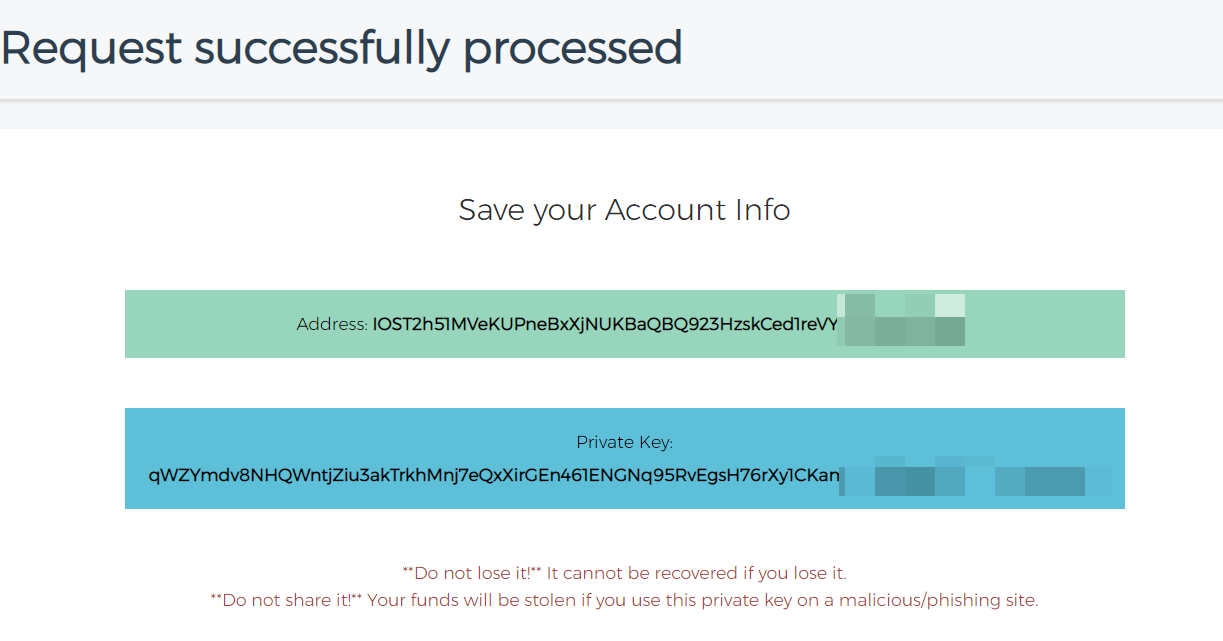
可以申请加入测试网络（可以看到其版本为0.5 Everest）：



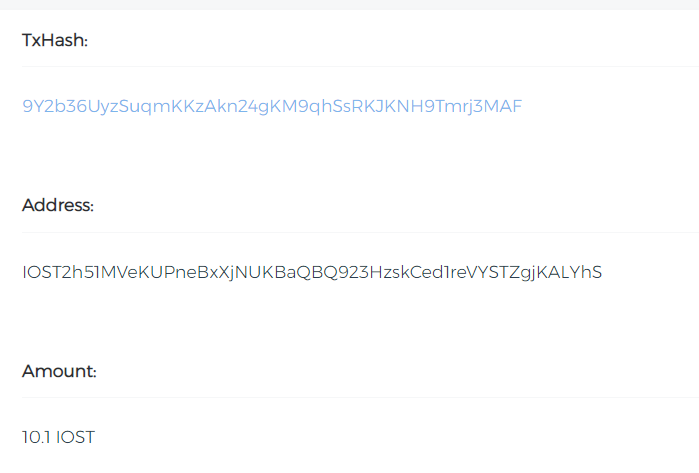
提交申请：



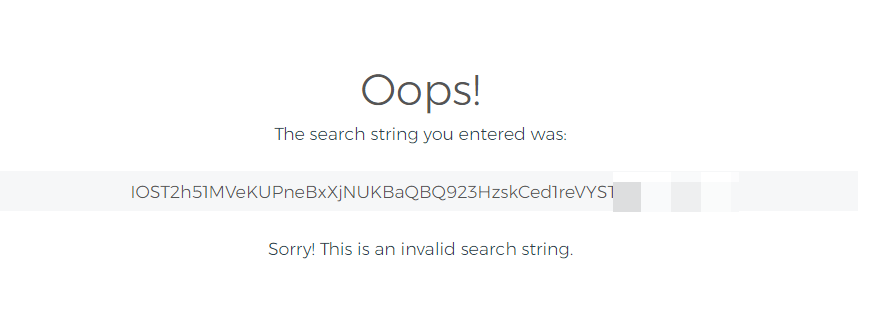
即可获得测试帐号：



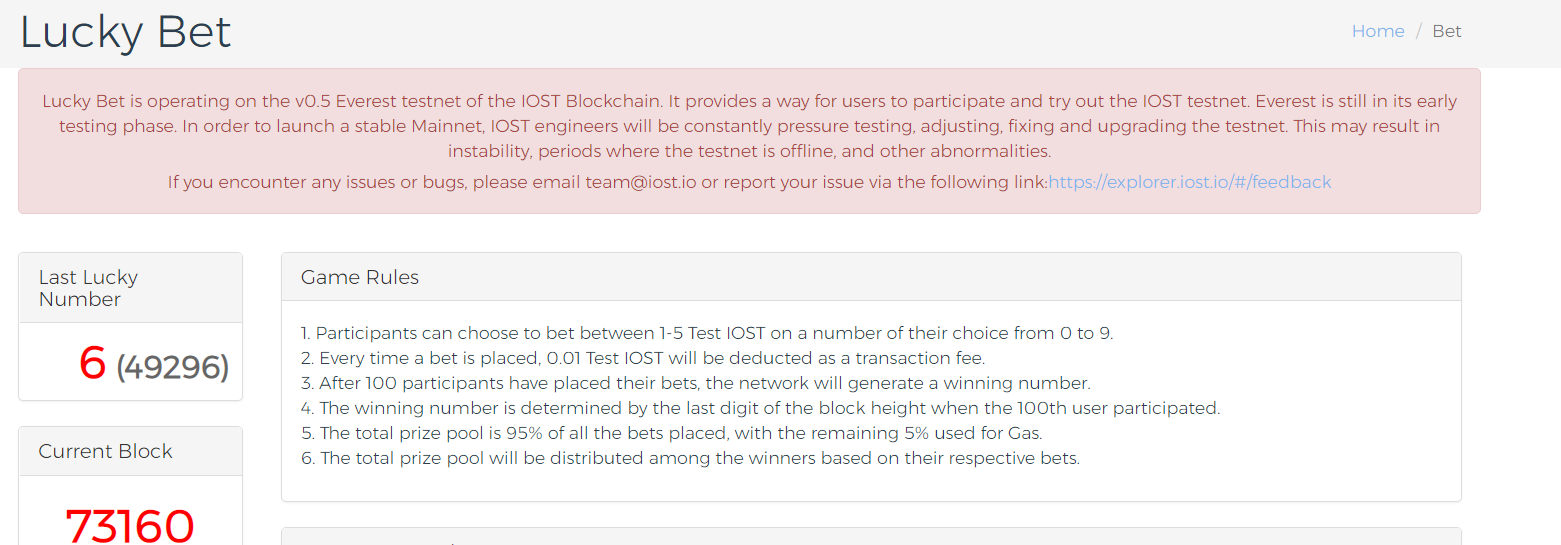
账号建立后，自动发送10.1 IOST：



但是查看ACCOUNTS页面，仍然只显示5个账号，交易记录也没有显示此笔交易。浏览器虽然提供了针对地址或者交易哈希值查询的功能，但是查询刚刚创建的测试地址，显示如下结果：



目前测试网络的主要测试手段，是提供了一个名为Lucky Bet的游戏。该游戏运行在0.5 Everest版本的IOST区块链上，帮助用户测试该网络。



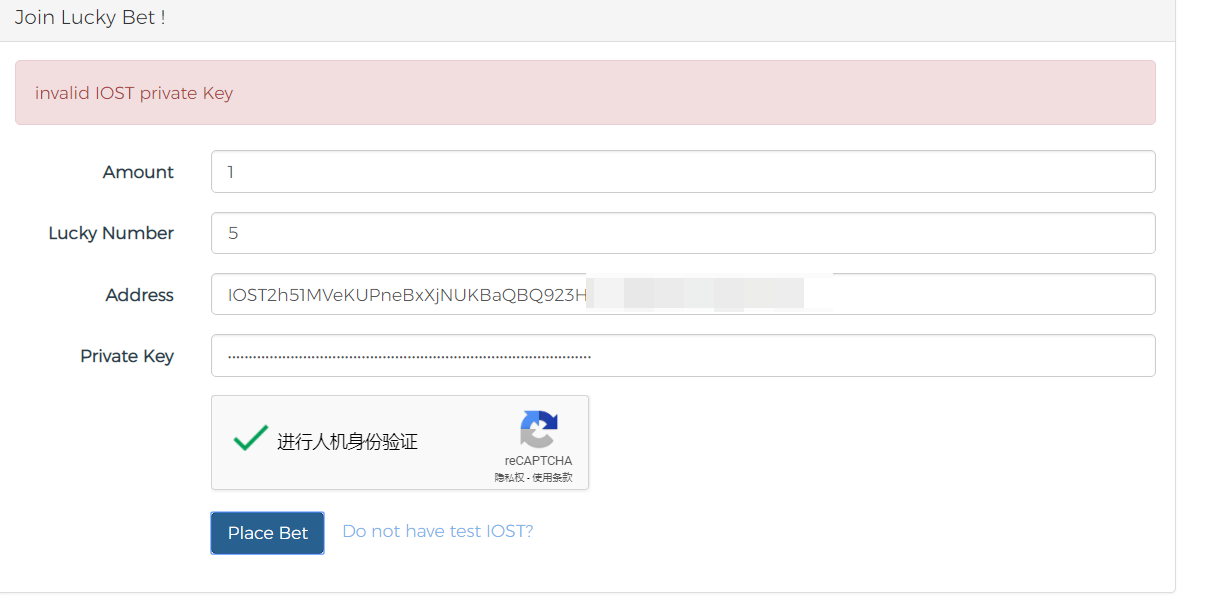
游戏规则：

1. 参加者可以针对他们选择的数字（1到9之间），投注1到5个测试IOST。
2. 每次投注之后，0.0.1个测试IOST会扣除，作为交易费。
3. 有100个成员投注之后，网络将会生成一个胜利号码。
4. 胜利号码取决于第100个用户加入时的区块高度的最后一位。
5. 所有投注总和的95%，作为奖金池。剩下5%用于gas费。
6. 总奖金将会根据获胜者（即猜对胜利号码的）的投注金额，在他们之间分配。

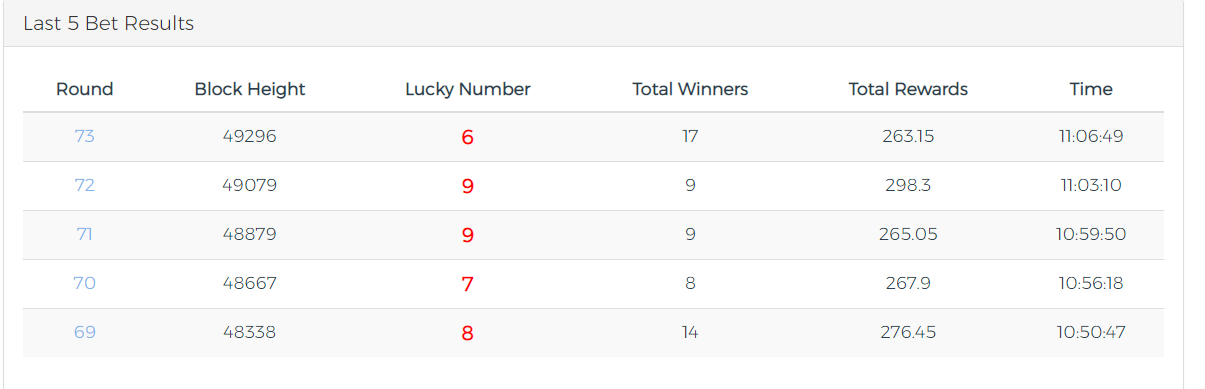
最近几次的投注结果如下：



选择投注，始终显示IOST密钥无效，工作不正常：



然后查看最近五次投注结果，譬如第73轮，显示区块高度为49296，所以幸运号码为结尾数字6，总共17个获胜者，总共奖励为263.15 个测试IOST。



但是点击73链接，可以看到详情，其中又显示“No Winner in this Round”。



Total Rewards的单位也跟列表中不一致。总之，测试网络的浏览器、游戏等显得缺点不少，完全处于初期开发阶段，很难使用。

# 2.代码质量评测

IOST的代码位于：<https://github.com/iost-official>， 关于区块链的核心代码位于：<https://github.com/iost-official/go-iost>，用GO语言，更新非常频繁。按照其文档说明，目前的测试网络版本Everest v0.5:

|  |  |
| --- | --- |
| 文档说明 | 实际测试结果 |
| 启动运行IOST共识机制PoB | 通过游戏表明测试网络已经启用 |
| 为用户提供测试网络代币 | 已经在上文中验证, 测试用户可以获得10.1个测试IOST |
| 存储和发送代币的钱包 | 没有找到验证方式，主页没有提供钱包。 |
| 能够运行IOST测试网络节点 | 提供了通过DOCKER建立节点的方式，可以运行。 |
| 显示交易和区块的区块链浏览器 | 主页上提供了浏览器，可以查询交易和区块，但是检索功能只能查到系统自动创建的地址，用户建立的测试地址无法查询。 |
| 建立私有的IOST测试网络 | 提供了通过DOCKER建立节点和网络的方式，可以运行。 |
| 支持密钥—值对存储、合同间API调用和多重签名的智能合约系统 | API调用方法如下：  --- main  -- your own comments  -- @gas\_limit 11  -- @gas\_price 0.0001  -- @param\_cnt 0  -- @return\_cnt 1  function main()  Put("hello", "world")  return "success"  end--f  --- sayHi  -- @param\_cnt 1  -- @return\_cnt 1  function sayHi(name)  return "hi " .. name  end--f  合约目前可以调用，也提供了相应的API，但是功能比较简单。 |
| 支持API的智能合约，以便迅速编写和部署代码 | 已经支持：  API列表：  Put(key, value) -> bool -- Write key-value pair to blockchain，only float，string are support  Get(key) -> bool, value -- Read value of a key, which written by contract itself  Transfer(from, to, account) -> bool -- IOST transfer，sender's signature should included  Call(ContractID, apiName, args) -> bool, value... -- inter-contract call, return API's returns  Deposit(from, value) -> bool -- deposit IOST to contract account  Withdraw(to, value) -> bool -- get IOST from contract account  Random(probability) -> number -- give probability and return a blockchain-random true/false result  Now() -> value -- return timestamp in seconds  Witness() -> string -- return current block's witness, in base 58 encode  Height() -> number -- return height of block  ParentHash() -> number -- return last byte of parent hash  ToJson(table) -> bool, jsonStr -- convert lua table to a json string  ParseJson(jsonStr) -> bool, table -- parse lua table from a json string |
| 用于在本地测试智能合约的测试“游乐场” | 已经提供  使用方法：  playground [-v init\_values.yml] [source\_code\_1][source\_code\_2]... |
| 运行在测试网络上的“幸运抽奖” 游戏 | 已经提供 |
| 实时监控和发送警报 | 没有提供 |
| 部署在实际网络环境中，运行在全球各地（东京、伦敦、新加坡、孟买、法兰克福和蒙特利尔）的多个节点上 | 没有证明 |

共识机制PoB代码已经开源，位于：<https://github.com/iost-official/go-iost/blob/master/consensus/pob/pob.go>，最新更新日期为两天前。

其他核心技术，例如高效分布式分片 EDS、原子协议 Atomix、微状态块 MSB均未在代码中体现。



# 3.总结

在5月份至今，IOST项目按照路线图取得了一些进展，包括对核心共识算法POB的代码进行了开源，上线了测试网络，并且让用户可以参与其中的Lucky Bet游戏。总体而言，有进展，但是问题还是不少，包括测试网络浏览器的功能不足、没有钱包，用户可以测试的功能非常有限；开源代码中高效分布式分片 EDS、原子协议 Atomix、微状态块 MSB等核心技术还未见踪影。