Multi blockchain, multi node network test plan

The Volentix team
sylvain@volentixlabs.com

October 13, 2020

1 Preparation

Multi blockchain test network
EOSIO

1. EOSIO Public Key
2. EOSIO Private Key
3. Cleos command
4. Path to nodeos binary
5. Path to keosd binary
6. Path to latest contracts directory
7. Path to 1.8.x contracts directory
8. Path to nodes directory
9. Path to genesis.json
10. Path to wallet directory
11. Path to log file
12. The eosio.system symbol
13. Max number of users. (0 = no limit)
14. Maximum user keys to import into wallet
15. How much funds for each user to spend on ram
16. Minimum stake before allocating unstaked funds
17. Maximum unstaked funds
18. Maximum number of producers. (0 = no limit)

19. Minimum producer funds
20. Number of producers for which each user votes
21. Number of voters
22. Number of users to transfer funds randomly
23. Time (s) to sleep to allow producers to sync
24. HTTP port for cleos
25. Killswitch
26. Unlock Wallet
27. Start boot node
28. Create system accounts (eosio.*)
29. Install system contracts (token, msig)
30. Create tokens
31. Set system contract
32. Initialiaze system contract
33. Create staked accounts
34. Register producers
35. Start producers
36. Vote for producers
37. Claim rewards
38. Proxy votes
39. Resign eosio
40. Replace system contract using msig
41. Random transfer tokens (infinite loop)
42. Show tail of node's log

ETHEREUM

1. Set up a distributed ethereum test network with openethereum.
2. Deploy the 777 contract onto it
3. create pools

4. Setup openzeppelin tests
5. bridging oracle/custodian contract

Docker network

1. Eos wallet
2. Openethereum
3. Bridging oracle
4. Bitcoin node
5. Vdex node

2 Tests

1. **Staking test**
   (a) v11111111111 stakes 10000 TVTX
2. **Persistency test**
   (a) Uptime
   (b) Less than 8 nodes
   (c) Register and unregister nodes
3. **Authority tests**
   (a) Open, unlocks eos wallet and signs executes oracle balance submission to EOS.
   (b) Register and unregister nodes
   (c) Reward test
      i. Test job selection
      ii. Test reward calculation
      iii. Test transfer
   (d) Oracle test
      i. Decouple eth-vtx oracle and uptime
      ii. Load tests

3 Conclusions/Todo

1. Bridging oracle on ethereum watching eos pool
2. Reverse proxy
   A Nginx HTTPS reverse proxy is an intermediary proxy service which takes a client request, passes it on to one or more servers, and subsequently delivers the server's response back to the client. In our case for key management keosd has to be launched as daemon behind reverse proxy(nginx) nginx will be used to enable password based authentication.
3. Ethereum oracle finance mechanism will be determined in next iteration
4. Bridging oracle persistency lock to be considered