

# HyperService 进展(190214~190220)

---

1. 安装并运行了 web3.js, 现在可以连接到 Geth 并返回信息。

在 js 这方面现在的问题是:

a. 现有的代码可能与此版本的 web3.js 的功能不一致, 可能原因是版本不一致, 等等。不过问题不大, 语句修改一下可以运行。

b. broker\_abi 没能 load, 原因待查。

c. web3.js 与 Geth 之间的交互成谜, 现在只能正常使用 Geth 的函数, 比如:

I. web3.eth.getAccounts(callback), 这时从 callback 可以得到内容。

II. web3.eth.defaultAccount, web3.eth.coinbase 这些固定的值不能读取, 不同步。

所以下一步会进一步寻找原因。

2. 把 GitHub 上现有支持 ethereum 的 python 库都看了一遍, 目前是 import 了 eth\_hash, eth\_rlp, Hexbytes。

3. 关于 merkle-proof 的 verify。先使用 getProof(address, keys, block), 可以返回对应 block, address 上的 storageproof。现在已经知道了:

a. storagehash 可以拉取到一个 storageMPT 的 node(storageroot)。

b. 关于 storageproof, 会返回对应 key 的值 value, 返回一个 rlp-serialized 的 node, 一个 storagehash 对应的 node, 一个路径 path 和对应 key 的值 value。

c. 用 path 匹配 storageMPT, 就可以验证 value。

4. 其他进展:

a. rlp 怎么应用到 Geth 返回的值已经搞清楚了。

b. leveldb 可以直接读取 ethdb(三个库 chain, lightchain, nodes)的内容

c. leveldb 现在有几个流行的库都分别了解了。非常不幸的是 leveldb 是 linux friendly 的, 并且有其他比如文件读取的历史问题, 所以暂时正在处理这方面的问题。

5. 总结:

a. 已经做了很多前置工作, 都不太顺利。现在已经进行到了倒数第二步。下面解决了 ethdb 读取的问题后只需写几个函数就能完成 verify。

b. 希望能够把一些东西封装一下, 以便以后使用。