需要的工具：

Scatte(钱包) <https://get-scatter.com/>

EosStudio <https://www.eosstudio.io/>

相关网站

浏览器 <https://bloks.io/>

Jungle3.0测试网 <https://api.monitor3.jungletestnet.io/#home>

BlockOne测试网 <https://testnet.eos.io/>

开发文档 <https://developers.eos.io/welcome/latest/reference/index>

EOSIO.CDT <https://github.com/EOSIO/eosio.cdt>

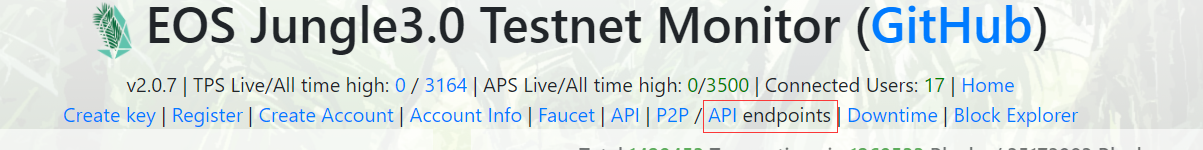
准备账号

<https://www.jianshu.com/p/9f9a6877a815>

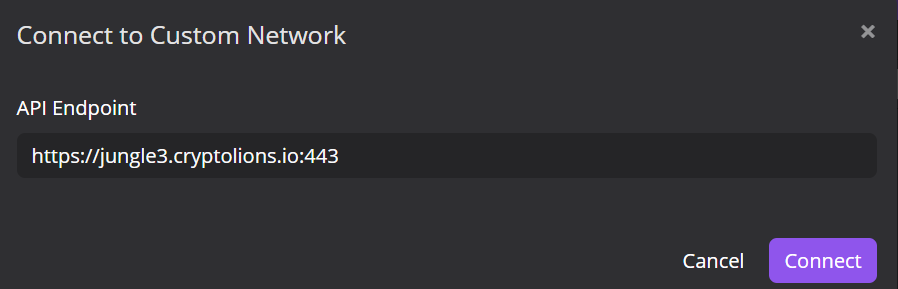
EosStudio正确的情况

选择网络

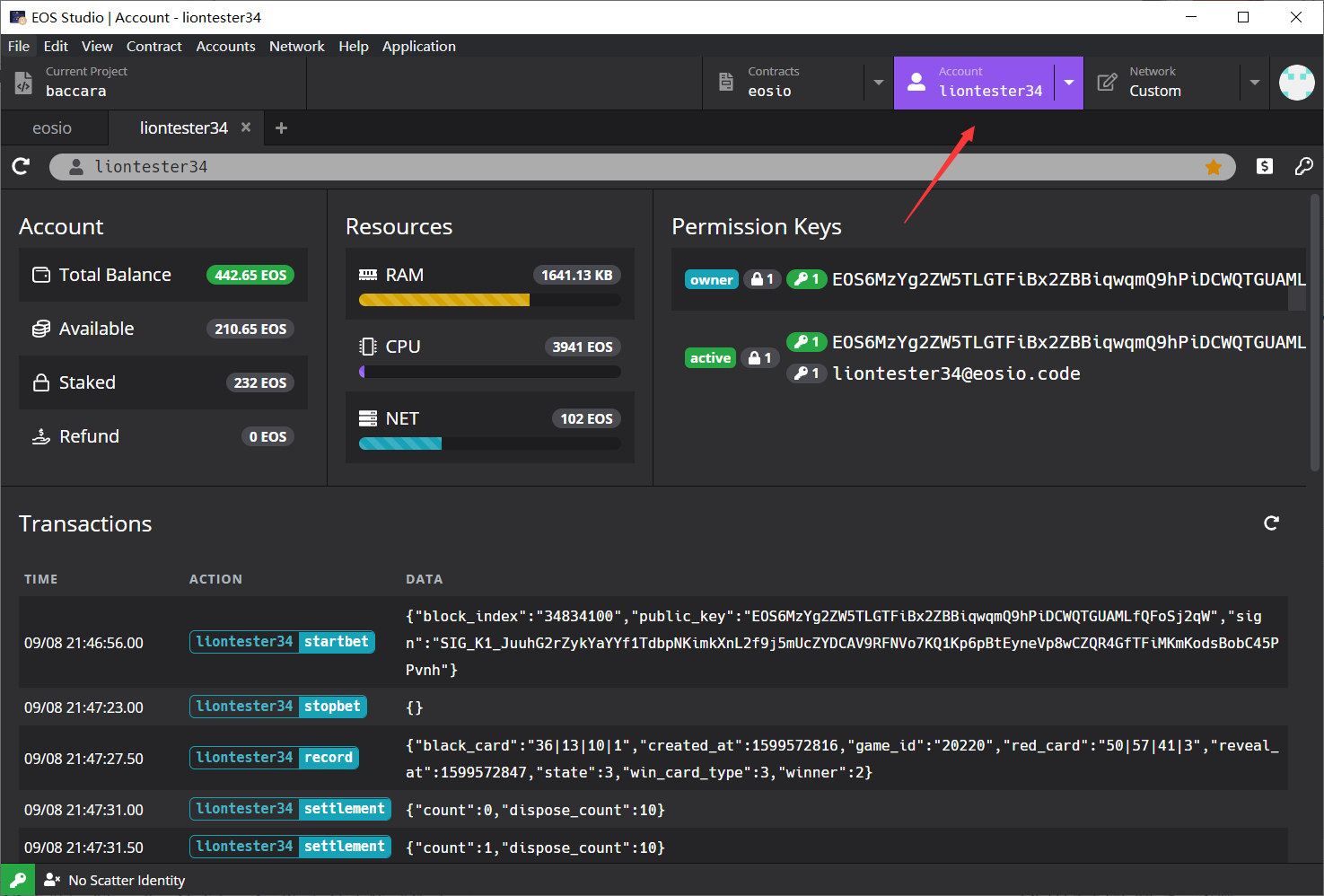
Custom: 需要api网址，一般可以在测试网络页面找到，比如jungle3的



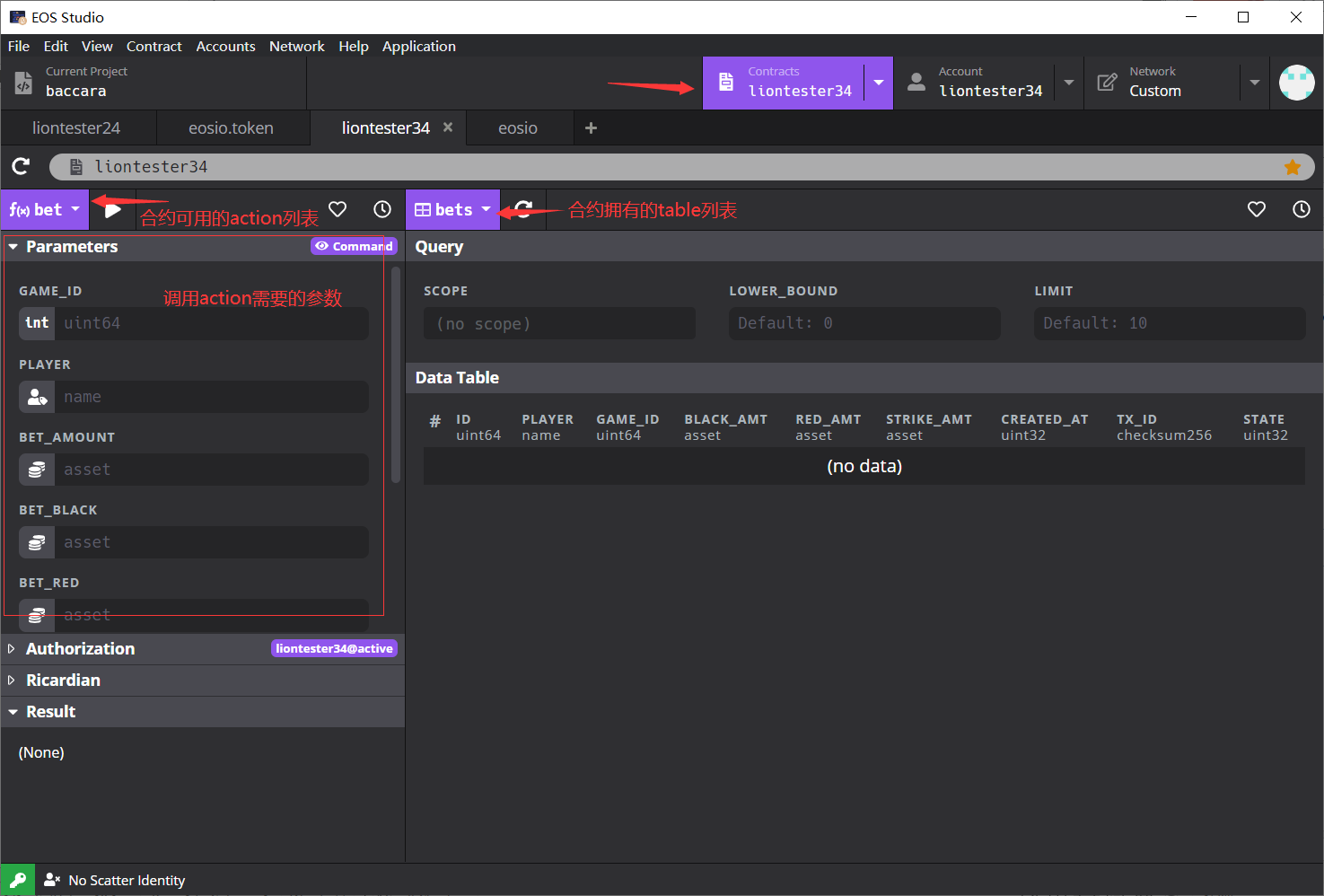
这里选的是： <https://jungle3.cryptolions.io:443>



查看账号

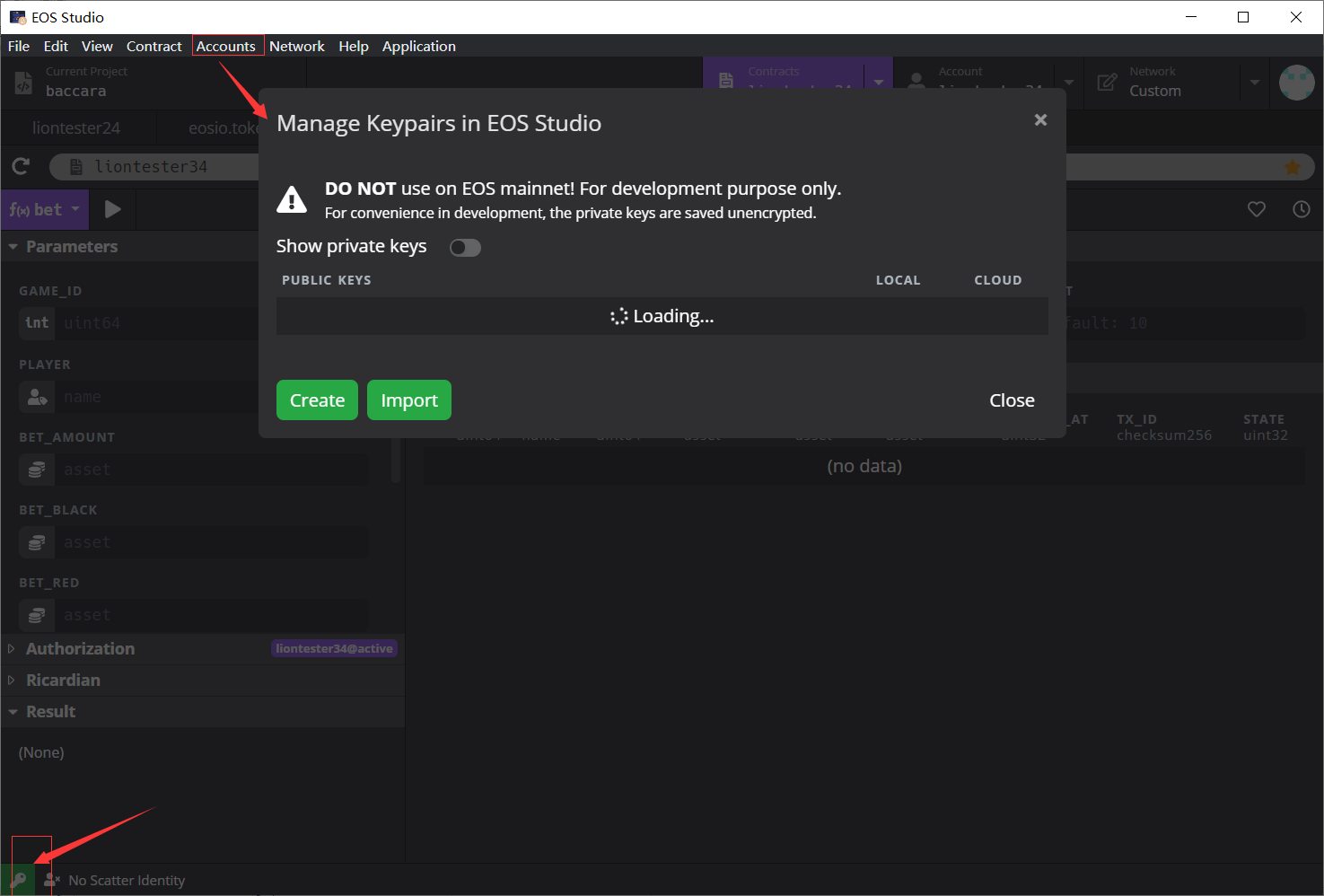


查看合约



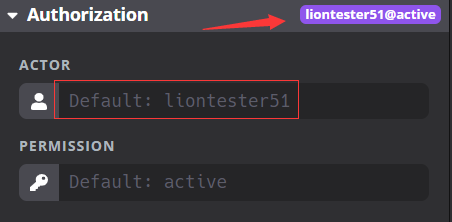
重点注意：

导入privatekey

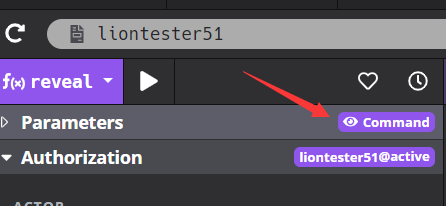


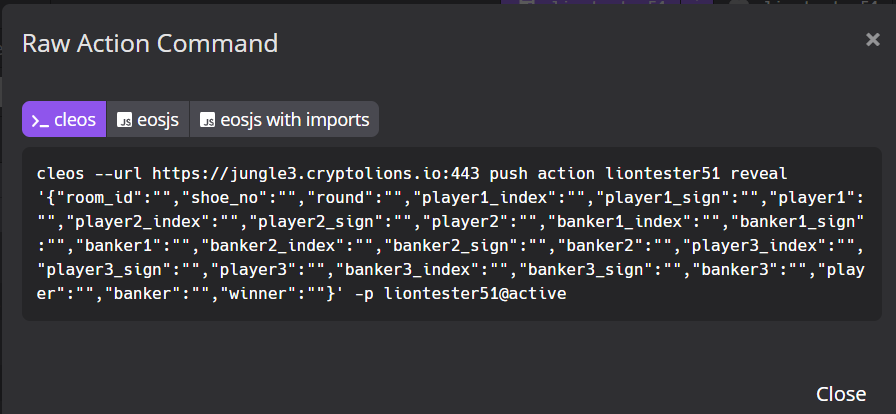
执行合约相关的action，是需要签名的，就需要用到privatekey，就是在这里找。相当于钱包。

执行合约的账号，默认就是当前选中的account，可以在authorization中填上要使用的账号信息。点击运行后，如果privatekey中找到了对应的私钥，就会正常执行。



执行的command是可以看到的：



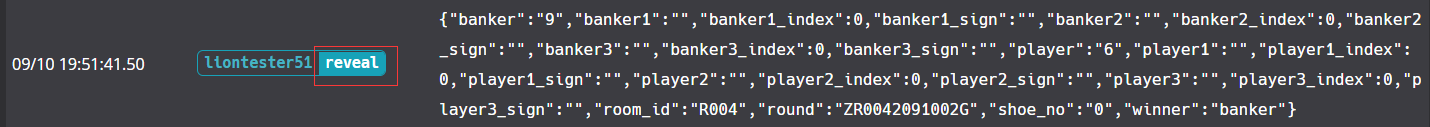


可以看到，实际也是使用cleos来实现的。

测试网络获取EOS

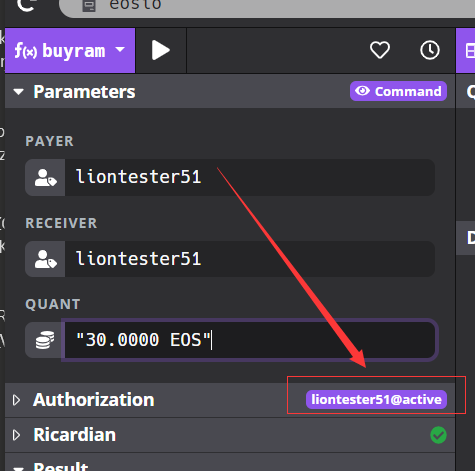
<https://api.monitor3.jungletestnet.io/#faucet>

Action: 就是智能合约提供的可调用方法(接口),参数可以作为上链数据使用。



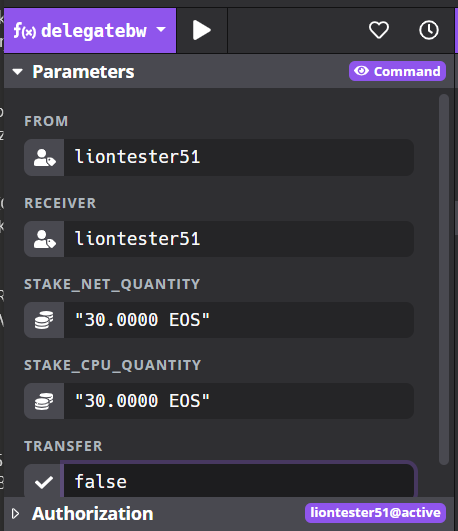
Eosio合约,这个是eos自带的基础合约中最重要的一个。常用的几个重要功能：修改权限，买RAM，抵押EOS获得NET和CPU等等。

buyram



这里有一个细节就是，eos的智能合约的action，应该是没有办法获取到是谁调用的action。所以使用参数传递调用者，就是最常用的方法了。

抵押EOS获取NET/CPU(使用的资源24小时后恢复): delegatebw

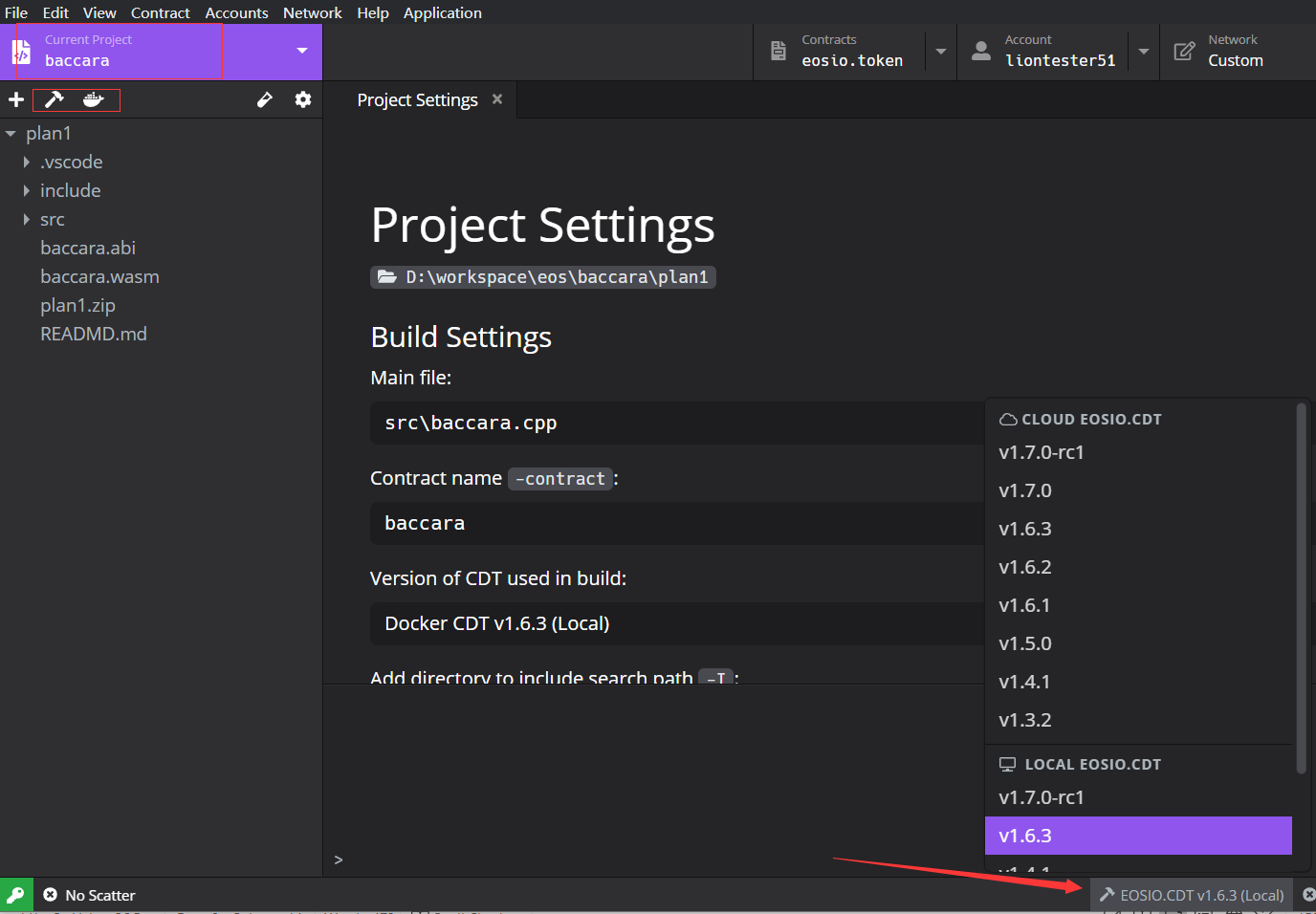


另一个比较重要的系统合约: eosio.token, 主要是转账等和代币(eos也是一种代币)相关的操作

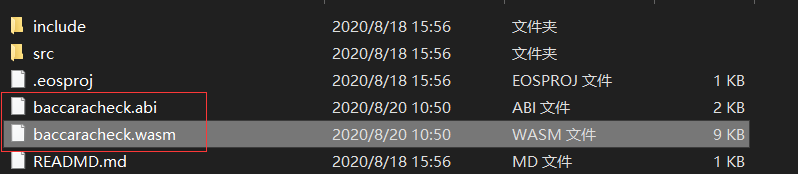
EOSIO.CDT

编译智能合约的工具，源代码中有个examples帶的几个例子比较好。

右下角可以选择编译的版本。



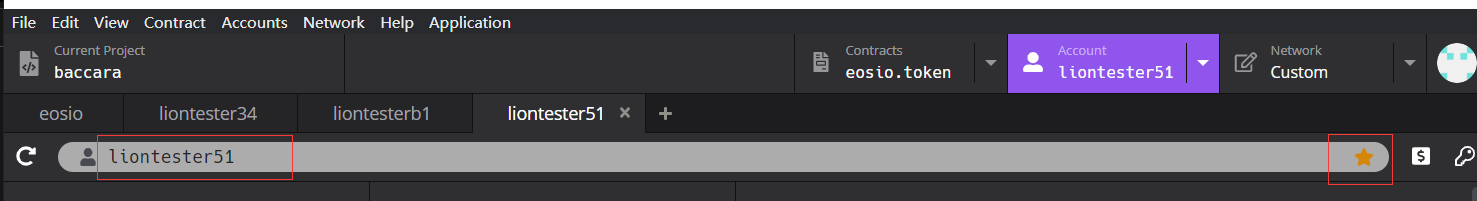
编译成功，会生成.abi和.wasm两个文件



编译的细节：

注意有可能出现编译成功，但是实际有警告，且没有生成wasm的情况。注意文件的生成时间。

部署时，可选的账号，是添加了收藏的账号列表



部署成功后，在合约的界面，就应该能搜索到部署的合约了(注意是账号名，而不是合约名)

也就是说一个账号和合约是一一对应的关系，两个等于是一体的。

cleos相关

创建账号：

Eos创建账号比较特殊，是需要有一个存在的账号去创建新的账号的。网络创建时默认会有eosio这个账号。这也是测试网创建账号需要专门的接口，主网账号比较贵的原因。

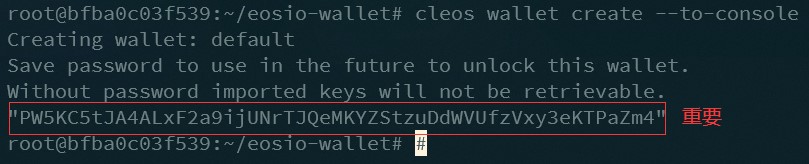
#cleos system newaccount

导入钱包

我的理解，钱包最重要的作用就是保存私钥，cleos命令需要签名时使用。钱包可以有多个，默认是default钱包。创建钱包时会返回一个key，非常重要，钱包软件的助记词就是用来记忆这个key的。

我们使用cleos连接第三方的api时，也是使用的cleos的本地钱包签名的。

#cleos wallet create --to-console       //一定记录下密码



##cleos wallet open //open|lock|unlock

#cleos wallet list //显示open状态的wallet

注意：

第一次启动和wallet相关的操作，会提示"/usr/opt/eosio/2.0.7/bin/keosd" launched。这时~/eosio-wallet目录被创建。也就是说,钱包的功能是依赖keosd的。

钱包的状态： 关闭|打开，lock|unlock。 使用钱包需要先打开钱包，在解锁钱包。

测试:

获取账号信息

cleos --url https://jungle3.cryptolions.io:443 get currency balance eosio.token liontester34 EOS

添加eosio.code权限

方法一: 使用block.io浏览器

方法而: cleos set account permission YOURCONTRACT active --add-code

语法解释：

require\_auth(get\_self()); 这个意思就是要求签名是合约的账号，也就是只有合约本身可以使用。

eosio::check(A < B, "A应小于B"); 如果A>=B, 调用action失败，并且错误中会看到A应小于B。

执行action的方法: