NEO补完计划

NEO的教程教智能合约的，教了大家两个概念，鉴权合约、应用合约。

从哪里开始？应用合约呀，因为鉴权合约很难走的通。

比如我们写了一个密码合约，只要知道密码就可以拿走这个账户里面的钱

## 密码账户：

编写了如下合约：

public static object Main(string method, object[] args)

{

if (Runtime.Trigger == TriggerType.Verification)

{

if (method == "0214")

return true;

}

return false;

}

他的avm如下，他是一个鉴权合约，不需要发布他。

55c56b6c766b00527ac46c766b51527ac4616168164e656f2e52756e74696d652e47657454726967676572009c6c766b52527ac46c766b52c3642a00616c766b00c30430323134876c766b53527ac46c766b53c3640e00516c766b54527ac4620f0061006c766b54527ac46203006c766b54c3616c7566

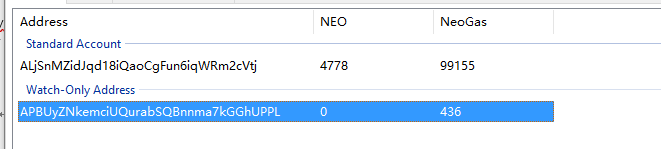
AVM的Hash是

0x86cf7371f5511257f682cc47be48563a3ff03f51

他的地址是

APBUyZNkemciUQurabSQBnnma7kGGhUPPL

对，就是这么算的，每一个脚本都有一个地址，不用发布，你往里面存钱就往地址转。

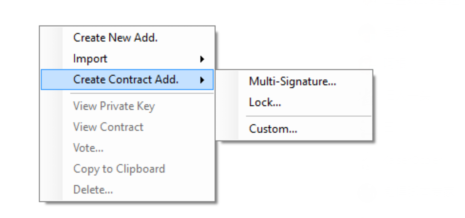


但是取钱，麻烦了。

我已经往这个地址放了一些GAS了，欢迎你用NEOGUI来取。

一试便知，很难。

多重签名合约和锁仓合约，NEOGUI直接支持了，比较容易。另外NEOGUI支持只有一个签名参数的自定义鉴权合约

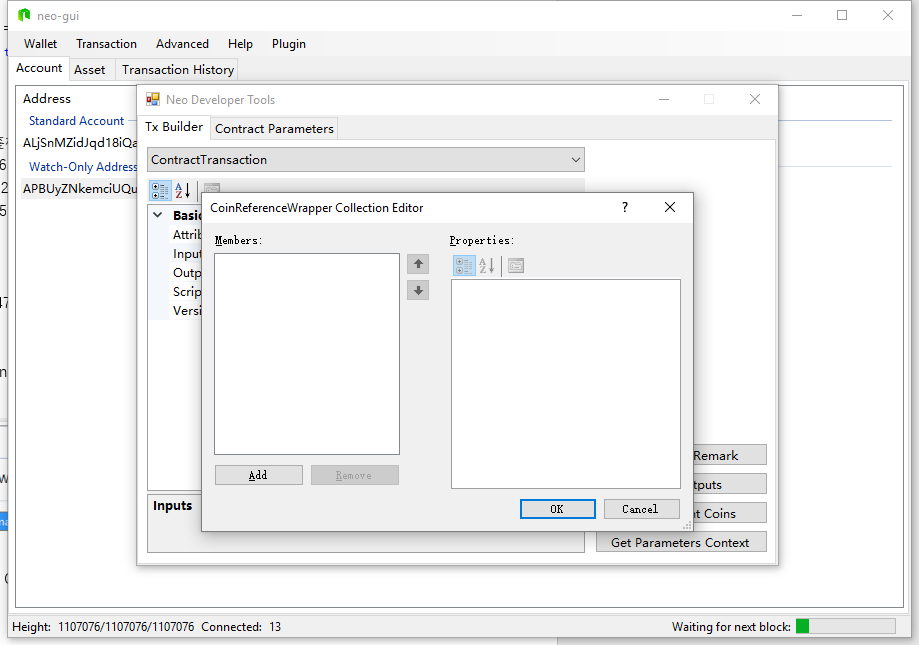


但鉴权合约一旦你用起来，肯定是想要自由奔放，那怎么办。

NEOGUI有没有办法取走，有。

要用F12 这个工具，自定义交易。

但是这个工具不太好用，我折腾了十分钟放弃了，于是我把这436个gas会一直留在那里，作为探索这个工具的奖励。



## 应该怎么办

要处理这样类型的交易，就需要你有开发钱包客户端的能力。你没听错，开发钱包客户端的能力。

具体一点，就是你需要理解 交易的 输入 输出 见证这几个概念，并且能自己拼出交易来。

DAPP 就是用人性化的界面面对用户，背后拼出交易。

也就是说，你连个钱包客户端都开发不了，做DAPP，那就洗洗睡吧。