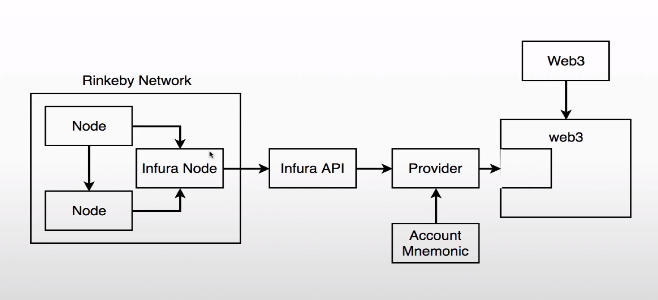
2019-03-22工作记录-部署infura

本文记录如何通过infura库把自己的合约部署到通用的测试网络Ropsten上面。

第一部分 工作原理

以前使用Remix在线浏览器写的合约能直接部署在测试网络Ropsten上面，原因是Remix太强大了，能直接连通测试网络。现在自己开发的代码项目工程不能再使用Remix了，如何搞呢？

结论是组合框架： Atom + solc + Web3 + mocha + ganache + infura + metamask



简要说下各组件功能：

Atom： 工程IDE，创建项目工程框架，编辑代码。

Solc： solidity文件编译器，对项目中的sol文件编译输出成二进制字节码和ABI接口。

Web3： 以太坊基金发布的 JavaScript 库 —— Web3.js，是一个通过RPC 调用 和本地以太坊节点进行通信的js库。web3.js也可以与任何暴露了RPC接口的以太坊节点连接。web3中提供了eth对象(web3.eth)来与以太坊区块链进行交互。所以这是一个真正的桥梁，能把你编译好的合约与以太坊网络连接起来。

Mocha：简单地说，方便测试我们自己的代码

mocha是JavaScript的一种单元测试框架，既可以在浏览器环境下运行，也可以在Node.js环境下运行。使用mocha，我们就只需要专注于编写单元测试本身，然后，让mocha去自动运行所有的测试，并给出测试结果。

mocha的特点主要有：

* 既可以测试简单的JavaScript函数，又可以测试异步代码，因为异步是JavaScript的特性之一；
* 可以自动运行所有测试，也可以只运行特定的测试；
* 可以支持before、after、beforeEach和afterEach来编写初始化代码。

Ganache：本地的测试网络，功能相当于私链。当编写好代码后可以在Ganache创建的内部网络中运行，创建出合约对象，调用方法等等，如果能通过就可以准备发布在以太坊上了。

Infura：与公链节点通信的代理服务器。为了能与公链通信，一般人们需要建立自己的节点，常常涉及1000G的下载存储，消耗太大了。如果不是为了挖矿，也不需要费劲建立这样的全节点。Infura就是这样的代理服务器，能把人们的请求发送到公链上。当然前提是你要在infura网站上注册一个自己的账号，以后的程序中将会嵌入这个账号，用这个账号提交给infura服务器。

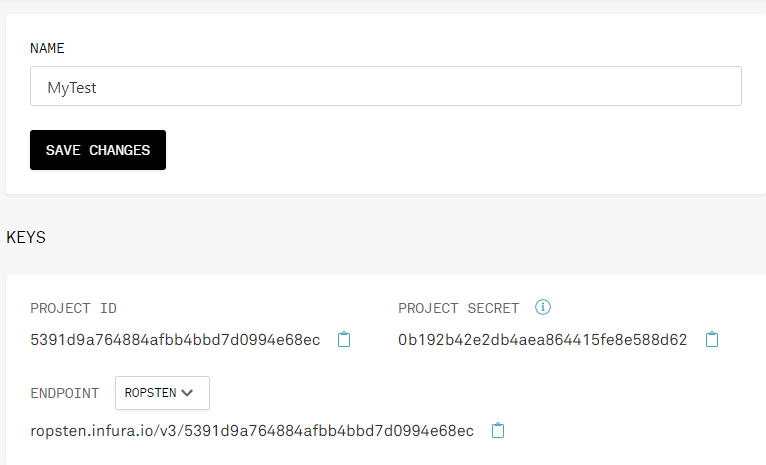
Metamask：这是大名鼎鼎的钱包了，存储了你的账户公钥和私钥。Metamask作为浏览器的扩展程序，能自动注入web3.js, 支持web3.js访问。当要把我们的合约发布到公链上时，需要指定用户身份，程序中要嵌入自己账户的公钥（12个助记词）

参考：https://www.imooc.com/article/67381

第二部分 注册Infura账户

2.1 网站：[www.infura.io](http://www.infura.io)

第一次访问该网站需要注册一个用户，要填写用户名、验证邮箱、密码。之后必须点击验证邮箱收到的激活邮件重新访问infura，生成该网站用户的私钥和公钥。



PROJECT ID： 5391d9a764884afbb4bbd7d0994e68ec

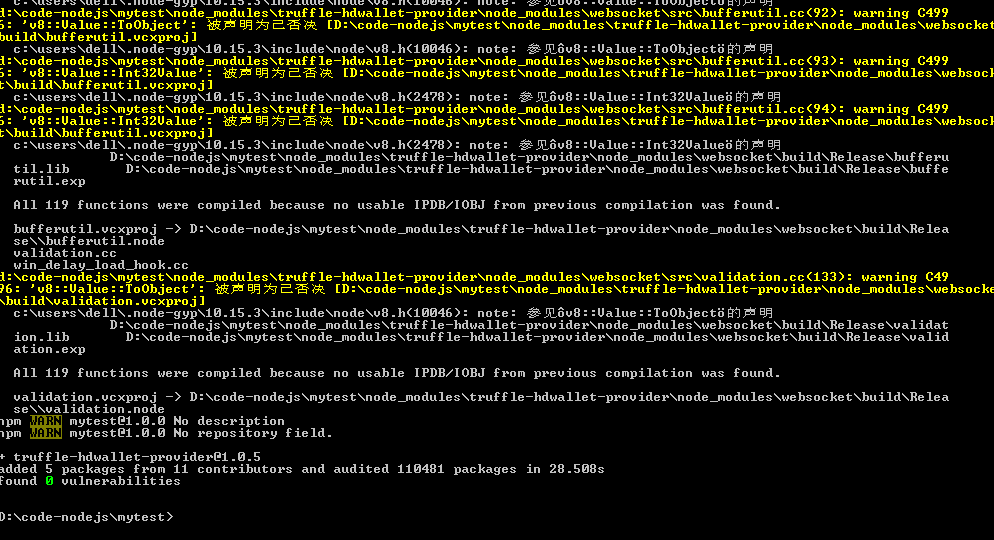
PROJECT SECRET：https://ropsten.infura.io/v3/5391d9a764884afbb4bbd7d0994e68ec

说明：这个工程账号地址非常重要，因为以后的合约命令是通过这个账号发送到infura网站，验证通过后才会转发到Ropsten测试网络中。我遇到的故障就是一旦调用send命令就石沉大海毫无反应，后来换用老师的账号就能成功，输出交易地址。我删除了原来的infura工程，重新建立了一个工程，使用新的工程地址运行成功了。

第三部分 终端安装 truffle-hdwallet-provider@1.0.5

在终端下执行：

D:\code-nodejs\mytest>npm install --save truffle-hdwallet-provider



第四部分 编写运行deploy.js

在工程中新建deploy.js，内容如下：

const HDWalletProvider = require('truffle-hdwallet-provider');

const Web3 = require('web3'); //大写的Web3是导入的类

// 推荐方法：

const {interface, bytecode} = require('./compile');

const provider = new HDWalletProvider(

'tribe warrior flush finger target hope unveil whip gloom popular guide unique',

'https://ropsten.infura.io/v3/5391d9a764884afbb4bbd7d0994e68ec'

);

//这里就是构造函数，通过infura把测试网络ropsten和web3库连接起来，生成一个对象小写的web3.

const web3 = new Web3(provider);

//定义匿名函数

const deploy = async ()=>{

console.log("========001============");

const accounts = await web3.eth.getAccounts();

console.log("========002============");

console.log('attempt to deploy contract', accounts[0]);

var result = await new web3.eth.Contract(JSON.parse(interface))

.deploy({data:'0x'+bytecode})

.send({from:accounts[0],gas:1000000});

console.log("========004============");

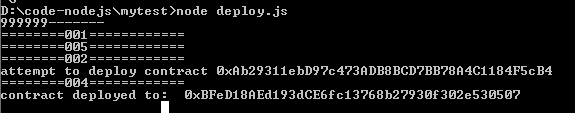
console.log("contract deployed to: ",result.options.address);

}

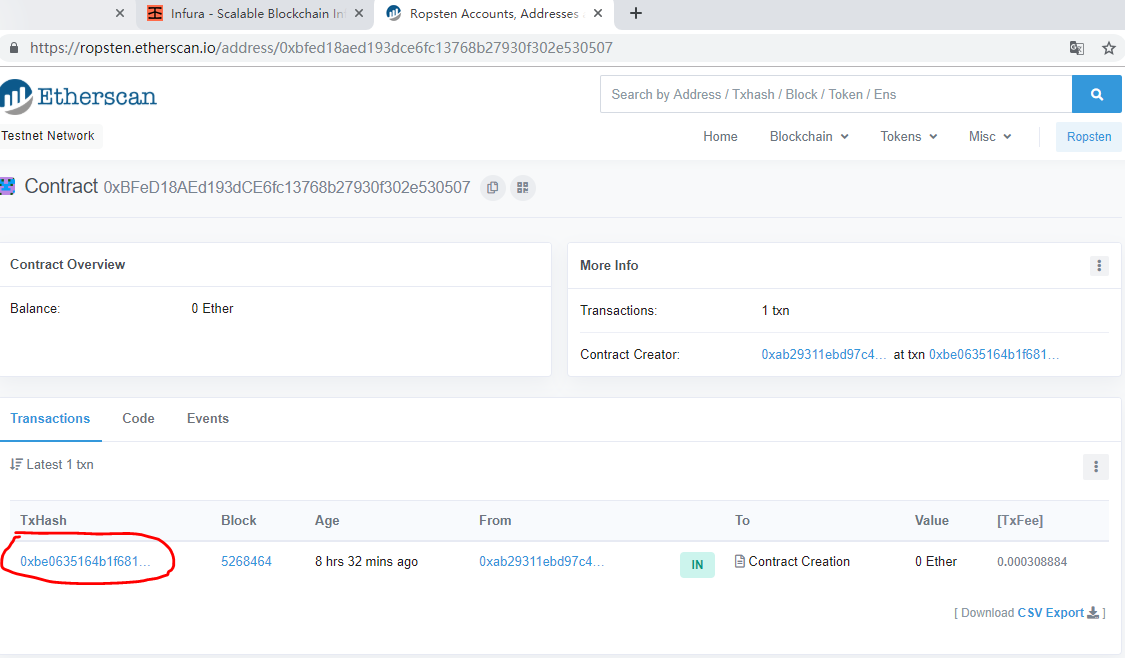
//执行这个函数

deploy();

运行结果如下图：



上面的地址就是发送的测试网络上交易地址，可以在[https://ropsten.etherscan.io](https://ropsten.etherscan.io/)网站上查看详细交易记录：



点击红圈查看详细信息：状态已经是Success， 表示被确认在了测试网络上。

