

Blockchain Setup Tips

Tim Clancy 2.21.2018

General Instruction Notes

To get started you're going to want a testing framework of some kind. I recommend [truffle](#) because it seems to be the most popular right now, so you won't have problems finding good resources for it. It lets you write scripts for deploying contracts to various Ethereum networks, and also write detailed tests. I have run into issues in the past with intermittent failure of complex test cases on Windows 10, so I would recommend using a different operating system. I currently use an Ubuntu 16.04 installation to develop.

Once you have truffle running, you have a lot of options for what software you use to run a local blockchain for testing. I would recommend [ganache-cli](#), it's pretty easy to use and flexible if you need more fine-tuning for tests. If you'd prefer a visual interface, you can also use [ganache](#).

Basically you run ganache to setup a local testing blockchain, then deploy your contract with truffle. You can run any tests you've written as well. You write your tests in JavaScript (or Solidity for unit testing, but that seems to have less documentation) and your actual contracts in Solidity. I found the truffle docs to be pretty good [resources](#).

I also found [this article](#) to be an exceptionally useful guide for writing more advanced tests.

Here's an okay [resource](#) for getting started with Solidity, but I think there's a lot more to be learned by looking through some of the known-good code provided by [OpenZeppelin](#).

And if you want to hack/experiment/deploy fast, and don't necessarily care about in-depth local testing, you might enjoy using an online IDE like [Remix](#) with [MetaMask](#) installed in your browser to handle your wallet. You can get test Ether for Kovan, one of the testing networks, by following the instructions [here](#).

Virtual Machine Image

I've created a [virtual machine image](#) which includes an Ubuntu 16.04 installation setup with a simple test example demonstrating everything described above. It will hopefully teach some familiarity with using ganache-cli, truffle, and deploying a simple contract to test. It also includes the appropriate testing libraries from OpenZeppelin and babel configuration to leverage better ES6 features like "await" which makes writing cleaner tests much easier. [Download](#) to try!

Contact Me

Hope this helps a bit, and if you have any questions or bump into snags then feel free to reach out! I'm available on the [Penn Blockchain Slack](#), at (508) 431-6847, or by [email](#).