

Testbed documentation

To be able to deploy smart contracts to the chain some changes had to be made to dockerfile. It turns out that the truffle folder with the contracts and migration files were never added to the dockerfile. This has now been done, so it is now located in the same level as the workspace folder. Also the workdir, in other words the directory that the image starts in, has been changed from “root” to “/”. This is done to make it easier to automate the whole process.

When smart contracts are deployed it is done with the truffle.js file in the truffle folder. This contains an address which is the address of account that is deploying the contracts.

That address is first known on runtime when the geth account is created. After that the address in the truffle.js file has to be changed to the new address. Because we want the testbed to run automatically, a script has been made that changes this address on runtime.

The python script made for this purpose first runs the command “geth account list --keystore workspace/miner1/keystore/” and saves the result in a file called “accountKeys.txt”. The command basically lists all accounts in a geth node.

After this the script extracts this account from the txt file and then inserts in the truffle.js file at the “from:” attribute.

For an account to be able to deploy contracts it has to be unlocked. This can of course first be done when the account has been created. It should be possible to do this through the --exec command in geth. For some reason this just wouldn't work. So the solution was to use the command “geth account --unlock 0 --password password.txt”. This command is part of the mine command.

The command simply unlocks the account at index 0 and then uses the password in the password.txt file. This is done through the “run.sh” bash script. This way the unlocking of the account has been automated and is done when the account has been created.

To be able to run the testbed with a single command a bash script “run.sh” has been made that executes all the commands and scripts mentioned in this paper.

It starts out by initializing the test chain and then creates an account. Then it unlocks this account and starts mining.

To make it easy to setup the testbed two files have been made.

One is the setup.py which locates the .env and then changes the VOLUME_PATH variable to the right path, so that it can run as intended.

Second is the setup.sh which installs all necessary software needed for running the testbed.

These files are located in the ethereum-testbed folder.

To run the testbed automatically the smart contracts have to be deployed automatically.

To do this a python script is used. The script uses Pexpect to interact with the Geth node, to monitor the balance of the Geth node. When there is enough ethers to deploy the contracts, the script quits and lets the “truffle migrate” command in the “run.sh” script run. This deploys the contracts and it is then possible to interact with the contracts. After the test scripts are loaded by the use Pexpect. Then the supplied testcases will be executed and the result is printed in the terminal.