Introduction to Bitcoin and Blockchain Technology

By Col. Wichit Saiklao, PhD.

Learn the concept of Bitcoin, the technology behind Bitcoin, the concept of decentralized trust, cryptography, consensus algorithm, blockchain, Ethereum and smart contract

Teaching Style: Explain and Hands on

Duration: 3-4 hours

Topics

Part I What is Bitcoin/Blockchain [1 hr]

- What is Bitcoin

- Why Bitcoin

- How Bitcoin works

- What is Blockchain

Part II Hands on tutorial on Blockchain [3 hrs]

01. How to install **Geth** client on Windows used for

- downloading the blockchain

- running an Ethereum node

- mining

- sending/receiving transaction

- contract

02. How to create a local private multi-node Ethereum network (geth console)

- initialize new blockchain

- connect Ethereum node in console

- start mining

- add client node to form a private Ethereum network

03. Compile and run your first smart contract against local private node

To create and deploy your first smart contract on your local private test network

- install the local copy of the browser-solidity compiler

- compile and deploy smart contracts on private Ethereum network

04. How to invoke your first smart contract from the **Geth** console (ABI)

To copy the ABI (application binary interface) from the browser compile window and access the contract by address directly from the geth console

05. Ethereum basics – accounts, contracts, nodes, miners

To summarize some of the concepts we discussed accounts, contracts, nodes, miners

06. Explained Ethereum Gas, price, limit explained (demo)

07. Using blockchain explorer to explore local blockchain data

---

NOTE ON WORKSHOP

01. Install geth (Go Ethereum)

1) Download Geth client from https://geth.ethereum.org/downloads/

2) Install geth

4) Test installation

Type “geth” in the terminal window

(CTRL-C to quit)

02. Create and test a private local Ethereum network

We will create a private local Ethereum network – with 2 nodes. We initialize a new blockchain from a sample genesis block. We will start using the Javascript console and some basic commands. We will learn and test the Ethereum protocol.

1) Setup the environment

- Create working directory ex. `C:\Users\mine\Ethereum`

- Create `genesis.json` file using your favorite editor in the working directory (`C:\Users\mine\Ethereum`)

```
{
    "config": {
        "chainId": 10,
        "homesteadBlock": 0,
        "eip155Block": 0,
        "eip158Block": 0
    },
    "coinbase" : "0x0000000000000000000000000000000000000000",
    "difficulty" : "0x400",
    "extraData" : "0x00",
    "gasLimit" : "0x8000000",
    "nonce" : "0x0000000000000042",
    "mixhash" :
"0x0000000000000000000000000000000000000000000000000000000000000000",
    "parentHash" :
"0x0000000000000000000000000000000000000000000000000000000000000000",
    "timestamp" : "0x00",
    "alloc" : {
    }
}
```

2) Initialise the block

```
geth –datadir "C:\Users\mine\Ethereum\local3" init
"C:\Users\mine\Ethereum\genesis.json"
```

3) start the console

```
geth --datadir "C:\Users\mine\Ethereum\local3" --ipcdisable console
2>console.log
```

4) create a 2nd node

```
geth –datadir "C:\Users\mine\Ethereum\local3-a" init
"C:\Users\mine\Ethereum\genesis.json"
```

5) Start on a different port

```
geth --datadir "C:\Users\mine\Ethereum\local3-a" --port 30304 console
2>console2.log
```

6) Geth the `admin.nodeInfo` enode from the 2nd instance and copy it into `admin.addPeer` in the first node

7) Key commands from the javascript console

`personal.newAccount()` – to create a new account

`miner.start(1)` – start mining

`eth.blockNumber` – current block height

`eth.getBlock(number).miner` – minor of block at that number

`eth.getBalance(account address)` – current balance of that account

03. Compile and run your first smart contract against local private node

We will install the local copy of the browser-solidity compiler so you can compile and deploy smart contracts against your own private Ethereum networks.

1) Install browser solidity

- Go to link https://github.com/ethereum/browser-solidity/tree/gh-pages

- Download the zip file and extract to local folder

- Open `index.html`

2) Write your first smart contract by copy and paste this into the window

```
pragma solidity ^0.4.0;
contract greeter {
	string greeting;
	function greeter(string _greeting) public {
		greeting = _greeting;
	}
	function greet() constant returns (string){
		return greeting;
	}
	function calculateProof(string document) constant returns (bytes32) {
		return sha256(document);
	}
}
```

3) Make sure geth is started

```
geth --datadir "C:\Users\mine\Ethereum\local3" --nodiscover --rpc --rpcport
"8545" --rpccorsdomain "*" console 2>console.log
```

4) Connect the browser to the Web3 Provider

- Choose the web3.provider and type http://localhost:8545

- Click “Create” to create the instance of the contract (make sure to unlock the account)

- Start miner if the miner not yet start

5) Key commands from the javascript console

`personal.unlockAccount(eth.account[0])` – to create a new account

`eth.pendingTransactions` – to view the list of pending transaction

04. How to invoke your first smart contract from the geth console (ABI)

This one we will copy the ABI (application binary interface) from the browser compile window and access the contract by address directly from the geth console itself.

```
pragma solidity ^0.4.0;

contract adder {

    string name;

    function setName(string _name) public {

        name = _name;

    }

    function getName() constant returns (string) {

        return name;

    }

    function add(int a, int b) constant returns (int) {

        return a+b;

    }

    function addAndRename(int a, int b, string _name) returns (int) {

        name = _name;

        return a+b;

    }

}
```

1) Copy the contents from the “interface” box in the browser-solidity window

2) Make sure that geth console is started

```
geth --datadir "C:\Users\mine\Ethereum\local3" --nodiscover --rpc --rpcport "8545" --rpccorsdomain "*" console 2>console.log
```

in the console type

unlock the account

```
>personal.unlockAccount(eth.accounts[0])
```

start miner

```
>miner.start(1)
```

copy ABI from the browser and paste

```
>var abi =
[{"constant":true,"inputs":[],"name":"getName","outputs":[{"name":"","type":"
string"}],"payable":false,"stateMutability":"view","type":"function"},{"const
ant":false,"inputs":[{"name":"a","type":"int256"},{"name":"b","type":"int256"
},{"name":"_name","type":"string"}],"name":"addAndRename","outputs":[{"name":
"","type":"int256"}],"payable":false,"stateMutability":"nonpayable","type":"f
unction"},{"constant":true,"inputs":[{"name":"a","type":"int256"},{"name":"b"
,"type":"int256"}],"name":"add","outputs":[{"name":"","type":"int256"}],"paya
ble":false,"stateMutability":"view","type":"function"},{"constant":false,"inp
uts":[{"name":"_name","type":"string"}],"name":"setName","outputs":[],"payabl
e":false,"stateMutability":"nonpayable","type":"function"}]
```

type abi to view the content of it

```
>abi
```
// it will show the content above

copy the address of the contract **0x3922188552213eabcbfbd88594586512386ec601**

```
>var address = "0x3922188552213eabcbfbd88594586512386ec601"

>var adder1 = eth.contract(abi).at(address)

>eth.defaultAccount = eth.accounts[0]
```
// set default account

Now we can access the function call of the contract (`adder1`) ex.

```
>adder1.getName

>adder1.setName("Hello Wichit")
```

**Note you have to ensure there is a miner running for the non-read only functions – since the state-changes have to mined into transaction blocks

**To check the result from the browser, it should show the same state

4) Key commands from the javascript console

`personal.unlockAccount(eth.account[0])` – to create a new account

`eth.pendingTransactions` – to view the list of pending transaction

`eth.defaultAccount` – to set default account

05. Ethereum basics – accounts, contracts, nodes, miners

---

06. Explained Ethereum Gas, price, limit explained (demo)

---

07. Using blockchain explorer to explore local blockchain data

1) Install **Git** (https://git-scm.com/book/en/v2/Getting-Started-Installing-Git) if you haven't already

2) Clone the repo

`git clone https://github.com/etherparty/explorer`

3) Download **Nodejs** and **npm** (https://docs.npmjs.com/getting-started/installing-node) if you don't have them

4) Change directory to 'explorer' then call this command

```
npm start
```

All dependencies will be automatically downloaded

5) Then visit http://localhost:8000 in your browser of choice. You might get an error message:

6) Install geth if you don't already have it, then run the this command

```
geth --datadir "C:\Users\mine\Ethereum\local3" --rpc --rpccorsdomain
"http://localhost:8000"
```

Then refresh the page in your browser

NOTE on Solidity

Data Type

- address
- uint
- string
- bool (true, false)
- mapping

Special words

- event
- modifier
- struct
- msg.sender
- msg.value
- throw