Lab 19-06-2023

[Note: If you have already downloaded the dependency libraries mentioned in Lab 16-06-2023, and your last command was “sudo apt-get install libssl1.0-dev” then please follow the steps below.]

[Disclaimer: Please do not rush, because if anything went wrong you would have to repeat the whole process.]

1. Start your VM and open the terminal and navigate to your desktop.

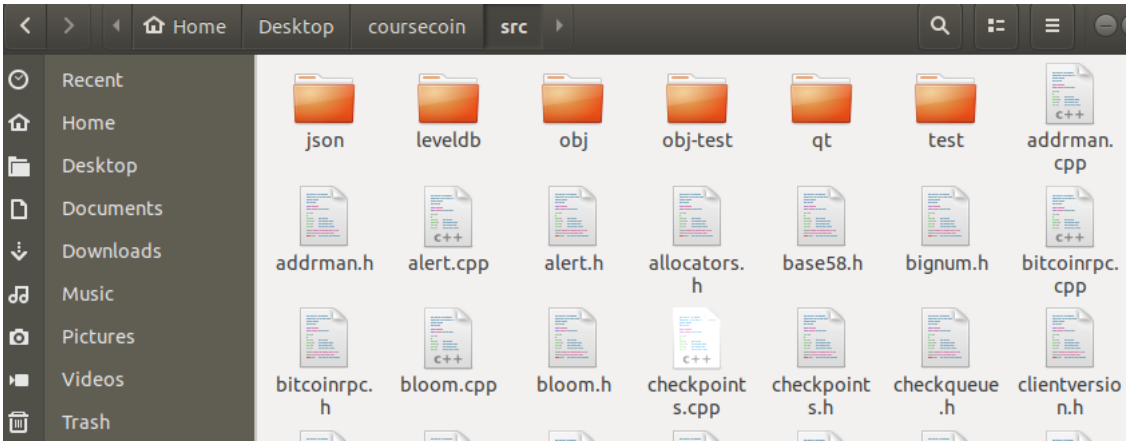
```
ubuntu@ubuntuVM:~$ cd Desktop/
ubuntu@ubuntuVM:~/Desktop$
```

2. Download the source code of Litecoin by typing the following command in the terminal of your Ubuntu VM.
   “git clone -b 0.8 https://github.com/litecoin-project/litecoin.git”

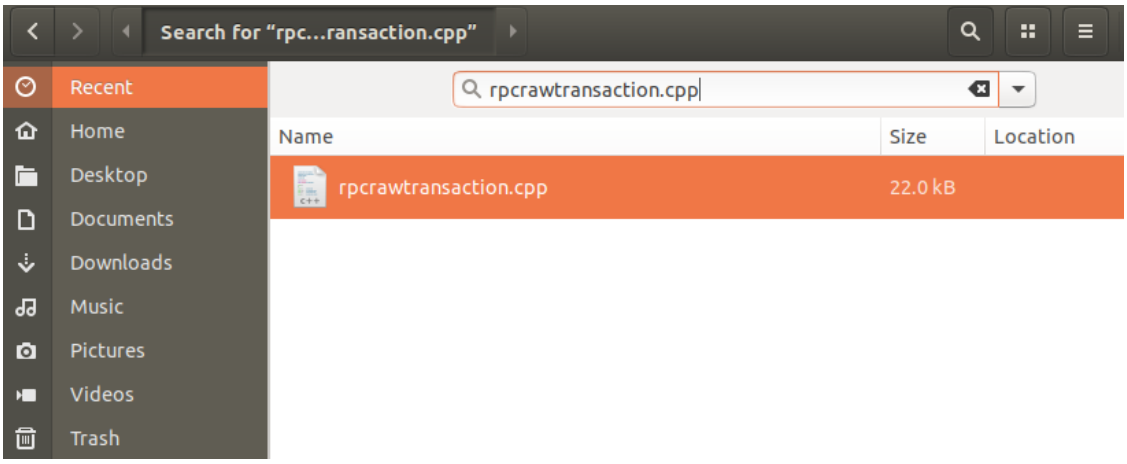
```
ubuntu@ubuntuVM:~/Desktop$ git clone -b 0.8 https://github.com/litecoin-project/litecoin.git
Cloning into 'litecoin'...
remote: Enumerating objects: 245912, done.
remote: Total 245912 (delta 0), reused 0 (delta 0), pack-reused 245912
Receiving objects: 100% (245912/245912), 223.59 MiB | 2.78 MiB/s, done.
Resolving deltas: 100% (173428/173428), done.
ubuntu@ubuntuVM:~/Desktop$
```

3. Right click and rename the Litecoin directory to the name of your liking. For example, coursecoin

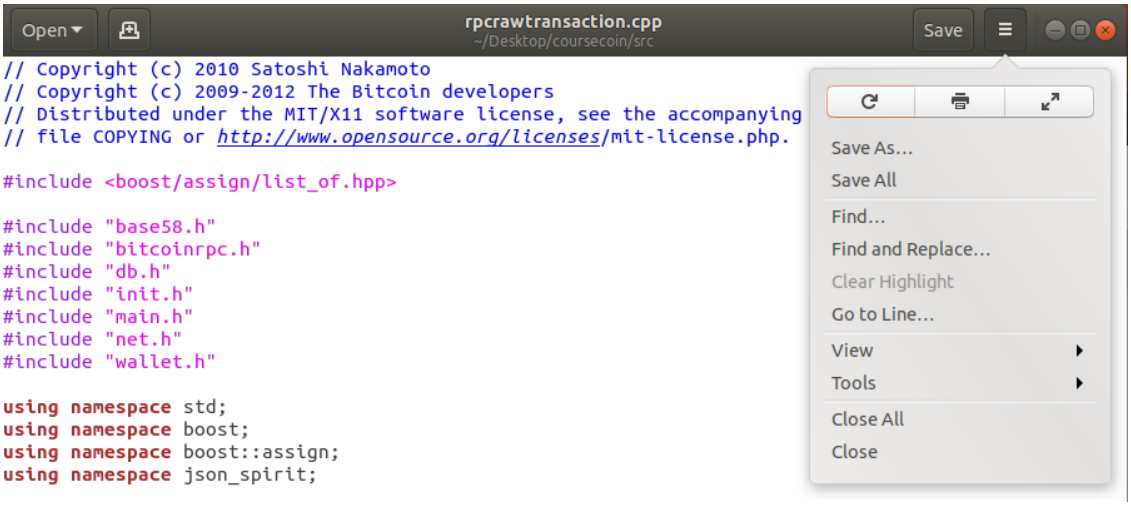
4. Navigate to inside of your coin directory, go inside the directory named “src”.



Search for a file named “rpcrawtransaction.cpp”



Open the file “rpcrawtransaction.cpp” and go to line “242”.



You'll see this snippet of code

```
            CTxDestination address;
            if (ExtractDestination(pk, address))
            {
                const CScriptID& hash = boost::get<const CScriptID&>(address);
                CScript redeemScript;
                if (pwalletMain->GetCScript(hash, redeemScript))
                    entry.push_back(Pair("redeemScript", HexStr(redeemScript.begin(),
redeemScript.end())));
            }
        }
```

Delete the "const" and "&" symbol to change the code as given below

```
        if (pk.IsPayToScriptHash())
        {
            CTxDestination address;
            if (ExtractDestination(pk, address))
            {
                const CScriptID& hash = boost::get<CScriptID>(address);
                CScript redeemScript;
                if (pwalletMain->GetCScript(hash, redeemScript))
                    entry.push_back(Pair("redeemScript", HexStr(redeemScript.begin(),
redeemScript.end())));
            }
```

Save the file and close it.

5. Now, head over to your terminal and navigate to your coin folder.

```
ubuntu@ubuntuVM:~$ cd Desktop/coursecoin/
ubuntu@ubuntuVM:~/Desktop/coursecoin$
```

Type the commands that are mentioned below in your terminal and instead of "coursecoin", "Coursecoin", "CourseCoin", "COURSECOIN" and "CCS", **you can use words corresponding to your own coin**.

i. **find . -type f -print0 | xargs -0 sed -i 's/litecoin/coursecoin/g'**
ii. **find . -type f -print0 | xargs -0 sed -i 's/Litecoin/Coursecoin/g'**
iii. **find . -type f -print0 | xargs -0 sed -i 's/LiteCoin/CourseCoin/g'**
iv. **find . -type f -print0 | xargs -0 sed -i 's/LITECOIN/COURSECOIN/g'**
v. **find . -type f -print0 | xargs -0 sed -i 's/LTC/CCS/g'**

```
ubuntu@ubuntuVM:~/Desktop/coursecoin$ find . -type f -print0 | xargs -0 sed -i 's/litecoin/coursecoin
/g'
ubuntu@ubuntuVM:~/Desktop/coursecoin$ find . -type f -print0 | xargs -0 sed -i 's/Litecoin/Coursecoin
/g'
ubuntu@ubuntuVM:~/Desktop/coursecoin$ find . -type f -print0 | xargs -0 sed -i 's/LiteCoin/CourseCoin
/g'
ubuntu@ubuntuVM:~/Desktop/coursecoin$ find . -type f -print0 | xargs -0 sed -i 's/LITECOIN/COURSECOIN
/g'
ubuntu@ubuntuVM:~/Desktop/coursecoin$ find . -type f -print0 | xargs -0 sed -i 's/LTC/CCS/g'
ubuntu@ubuntuVM:~/Desktop/coursecoin$
```

6. Now in your terminal navigate to the src directory inside your coin directory using the command “cd”. And type the command “make -f makefile.unix” to check whether all the changes that you made to your coin directory works or not.

```
ubuntu@ubuntuVM:~/Desktop/coursecoin$ cd src
ubuntu@ubuntuVM:~/Desktop/coursecoin/src$ make -f makefile.unix
Building LevelDB ...
make[1]: Entering directory '/home/ubuntu/Desktop/coursecoin/src/leveldb'
g++ -I. -I./include -fno-builtin-memcmp -pthread -DOS_LINUX -DLEVELDB_PLATFORM_POSIX -O2 -pthread -Wa
ll -Wextra -Wformat -Wformat-security -Wno-unused-parameter -g -DBOOST_SPIRIT_THREADSAFE -D_FILE_OFFS
ET_BITS=64 -I/home/ubuntu/Desktop/coursecoin/src -I/home/ubuntu/Desktop/coursecoin/src/obj -DUSE_UPNP
=0 -DUSE_IPV6=1 -I/home/ubuntu/Desktop/coursecoin/src/leveldb/include -I/home/ubuntu/Desktop/courseco
in/src/leveldb/helpers -DHAVE_BUILD_INFO -fno-stack-protector -fstack-protector-all -Wstack-protector
 -U_FORTIFY_SOURCE -D_FORTIFY_SOURCE=2  -c db/builder.cc -o db/builder.o
```

[Note: It will take a while so do not interrupt it.]

7. After the compilation has been done type “ls” in your terminal. If you get a file with a “d” at the end of the name of your coin directory, for example in our case the directory name was **coursecoin** so a file named **coursecoind** appeared after the compilation.

```
ubuntu@ubuntuVM:~/Desktop/coursecoin/src$ ls
addrman.cpp       compat.h      leveldb.cpp          obj-test              sync.cpp
addrman.h         coursecoind   leveldb.h            protocol.cpp          sync.h
alert.cpp         crypter.cpp   limitedmap.h         protocol.h            test
alert.h           crypter.h     main.cpp             qt                    threadsafety.h
allocators.h      db.cpp        main.h               rpcblockchain.cpp     txdb.cpp
base58.h          db.h          makefile.linux-mingw rpcdump.cpp           txdb.h
bignum.h          hash.cpp      makefile.mingw       rpcmining.cpp         ui_interface.h
bitcoinrpc.cpp    hash.h        makefile.osx         rpcnet.cpp            uint256.h
bitcoinrpc.h      init.cpp      makefile.unix        rpcrawtransaction.cpp util.cpp
bloom.cpp         init.h        mruset.h             rpcwallet.cpp         util.h
bloom.h           json          netbase.cpp          script.cpp            version.cpp
checkpoints.cpp   key.cpp       netbase.h            script.h              version.h
checkpoints.h     key.h         net.cpp              scrypt.cpp            wallet.cpp
checkqueue.h      keystore.cpp  net.h                scrypt.h              walletdb.cpp
clientversion.h   keystore.h    noui.cpp             scrypt-sse2.cpp       walletdb.h
coincontrol.h     leveldb       obj                  serialize.h           wallet.h
ubuntu@ubuntuVM:~/Desktop/coursecoin/src$
```

8. Close the VM, and remember to save the machine state.