Catapult Server – How to Build and Run on macOS

Overview

I hope others find this documentation to build and run Catapult Server on macOS useful.

Credits

This

1. The catapult build script used in this repo is based on “add build shell like eosio_build.sh (for mac) #7” by t10471 https://github.com/nemtech/catapult-server/pull/7

2. Nemesis block creation steps in this doc is based on ISARQ www.isarq.com Creation of the Nemesis Block in Slackware http://isarq.com/wp-content/uploads/2018/06/catapult-episode-2-nemesis-english.pdf

NEM Catapult Server

https://github.com/nemtech/catapult-server

Build Environment

macOS High Sierra (version 10.13.6)

MacBook Pro

Processor 2.3 GHz Intel Core i5

Memory 8 GB 2133 MHz LPDDR3

251 GB Flash Storage

Build Duration

About 2 hours

Dependencies and Versions

Library/Executable Name	Ubuntu 18.04	macOS
gcc & g++	7	n/a - macOS uses clang
cmake	3.11.1	3.11.1
boost	1.65.1	1.65.1
googletest	1.8.0	1.8.0
rocksdb	5.12.4	Latest
mongodb	Latest	Latest
zeromq/libzmq	4.2.3	4.2.5
zeromq/cppzmq	4.2.3	4.2.3
mongo-c-driver	1.4.3	1.4.2
mongo-cxx-driver	3.0.2	3.0.2

Clone this repo (fullcircle23/catapult-server-macos)

```
cd ~/<local-root>
git clone https://github.com/fullcircle23/catapult-server-macos.git
cd catapult-server-macos
```

Clone nemtech/catapult-server repo

```
cd ~/<local-root>
git clone https://github.com/nemtech/catapult-server.git
cd catapult-server
```

Copy macOS build script from fullcircle23/catapult-server-macos to nemtech/catapult-server

```
cp ~/<local-root>/fullcircle23/catapult-server-macos/catapult_server_build.sh .
```

Execute the macOS build script from inside nemtech catapult-server directory

```
./catapult_server_build.sh
```

Post Build Steps

Try running catapult.server as follows:

```
cd ./_build/bin
./catapult.server
```

I encountered the below 2 errors and have documented a resolution/workaround.

Error (i)

```
… No rule to make target '/usr/local/lib/libboost_<blah>', needed by
'catapult.server'.
```

If you get missing Boost error like the above then you'll need to install the missing Boost libraries by running `install-boost-lib.sh` script in ~/<local-root>/catapult-server/_build_dependencies/boost/1.65.1/lib. First copy the file from this repo and review and update the script if necessary, before running it. You may need to use sudo.

```
cd ~/<local-root>/catapult-server/_build_dependencies/boost/1.65.1/lib
cp ~/<local-root>/fullcircle23/catapult-server-macos/install-boost-lib.sh .
./install-boost-lib.sh
```

Now try running catapult.server again. If you are still getting similar missing library errors then you'll need to find that library and copy it to the appropriate location as specified in the error line. In many cases, the required library may already exists but under a different name and you would only need to create a symbolic link to it.

Error (ii)

```
… Missing libidn2.0.dylib library
```

If you get a "missing libidn2.0.dylib library" error and you find that libidn2.4.dylib exists, you could try creating a symbolic link as follows:

```
ln -s /usr/local/opt/libidn2/lib/libidn2.4.dylib /usr/local/opt/libidn2/lib/libidn2.0.dylib
```

Once catapult.server is able to run without complaining about some missing library, we are ready to proceed to the next step which is to create the Nemesis block.

Create the Genesis Block (Nemesis) for a New and Clean Cryptocurrency

1. Working directory

```
cd catapult-server
mkdir _build
cd _build
```

2. Copy configuration files

```
rm -r resources
cp -r ../resources .
cp ../tools/nemgen/resources/mijin-test.properties resources/
```

3. Generate 10 main accounts. You can generate less or more depending on your needs and resource limitations by changing the *-g* option value.

```
cd catapult-server/_build/bin
./catapult.tools.address -g 10 -n mijin-test > ../catapult.address.txt

head -n11 ../catapult.address.txt

--- generating 10 keys ---
private key:
8A28DA1BFB2E3BD71F063478F54D9AB80B8EDD71781488F20515434A65E273D4
public key:
B0D9E3C35AB3959E272F5E86E31495B9AE869AFB2902112F3D67C5F07F56ECAA
address (mijin-test): SCXD5QAN3W3FDZ5XQOWX7B7QF6AQOQWE3RJT6GBP
```

4. Edit the configuration file of the Nemesis block

```
vi ../resources/mijin-test.properties
```

5. Replace the lines marked in red with the green lines, as shown in the figure.

```
[cpp]
-cppFileHeader = ../HEADER.inc
+cppFileHeader = ../../HEADER.inc

[output]
-cppFile = ../tests/test/core/mocks/MockMemoryBasedStorage_data.h
+cppFile = ../../tests/test/core/mocks/MockMemoryBasedStorage_data.h

[distribution>nem:xem]

-SAAA244WMCB2JXGNQTQHQOS45TGBFF4V2MJBVOUI = 409'090'909'000'000
+SCSBPEXYDODOFC4LHR27KDVKRELXMRERKO4ZPDYV = 409'090'909'000'000
 SAAA34PEDKJHKIHGVXV3BSKBSQPPQDDMO2ATWMY3 = 409'090'909'000'000
 SAAA467G4ZDNOEGLNXLGWUAXZKC6VAES74J7N34D = 409'090'909'000'000
 SAAA57DREOPYKUFX4OG7IQXKITMBWKD6KXTVBBQP = 409'090'909'000'000
```

Note: The line SCSBPEXYDODOFC4LHR27KDVKRELXMRERKO4ZPDYV corresponds to the first account in the **catapult.address.txt** file, as shown in Step 3.

6. Create sub-directories in _build

```
mkdir -p seed/mijin-test/00000
dd if=/dev/zero of=seed/mijin-test/00000/hashes.dat bs=1 count=64
mkdir data tmp
cd tmp
```

7. Generate the Nemesis block

```
../bin/catapult.tools.nemgen --nemesisProperties ../resources/mijin-
test.properties
```

Tip: To have both stderr and output *displayed on the console* ***and*** *in a file*:
SomeCommand 2>&1 | tee SomeFile.txt

```
cd ..
cp -r seed/mijin-test/* data/
```

8. Start the Catapult server

```
cd bin && ./catapult.server
```

--- End of First Part ---