<https://paris-fire-brigade.github.io/data-challenge/post/2019/06/13/06-set-up-an-osrm-server-on-ubuntu.html>

sudo apt-get update

Install dependencies

sudo apt-get install build-essential git cmake pkg-config \

sudo apt-get install libbz2-dev libxml2-dev libzip-dev libboost-all-dev \

sudo apt-get install lua5.2 liblua5.2-dev libtbb-dev

Create a dedicated directory and move into to keep everything tidy and clean

mkdir osrm

cd osrm

Get OSRM from the official GitHub repo

git clone <https://github.com/Project-OSRM/osrm-backend.git>

Compile and install OSRM binaries

cd osrm-backend

mkdir -p build

cd build

cmake ..

cmake --build .

sudo cmake --build . --target install

The map pre-processing is quite memory intensive. For this reason, OSRM uses a library called STXXL to map its internal operations on the hard disk. STXXL relies on a configuration file called .stxxl, which lives in the same directory where you are running your software, to determine how much space is dedicated to the STXXL data structures.

Configure STXXL with an .stxxl file in your osrm folder.

cd ~/osrm

vi .stxxl

Write in it

disk=/tmp/stxxl,10G,syscall

Save and close it.

Because the speed profile script might depend on some Lua functions defined in the profiles library, we also create a symbolic link to it in the same directory by running the following two commands.

ln -s osrm-backend/profiles/car.lua profile.lua

ln -s osrm-backend/profiles/lib

Get openstreetmap data

wget <https://download.geofabrik.de/asia/pakistan-latest.osm.bz2>

wget <https://download.geofabrik.de/asia/jordan-latest.osm.bz2>

wget https://download.geofabrik.de/asia/malaysia-singapore-brunei-latest.osm.bz2,

bzip2 -d pakistan-latest.osm.bz2

bzip2 -d malaysia-singapore-brunei -latest.osm.bz2

Pre-process the .osm file with the car profile

osrm-extract pakistan-latest.osm -p osrm-backend/profiles/car.lua

Build the contraction hierarchy

osrm-contract pakistan-latest.osrm

To keep the osrm-routed process running after ending ssh session, use tmux

tmux

Start a routing engine HTTP server on port 5000 inside the tmux session

tmux

Leave/detach the tmux session by typing Ctrl+b and then d

You have now a high performance routing engine up and running.

Then If it directly not works change apache2.conf file and then change

ProxyPass /osrm http://localhost:5000

LoadModule proxy\_module modules/mod\_proxy.so

LoadModule proxy\_http\_module modules/mod\_proxy\_http.so

https://evrimo.com/osrm/table/v1/driving/67.112457,24.882638;67.113009,24.882293;67.111032,24.881284?annotations=distance

osrm-routed --port <portnum>

Step1:

Tmux new –s sessionname

Step2:

osrm-routed –port # map\_name.osrm

http://evrimo.com/dubai\_saudia/table/v1/driving/46.680113,24.715553;46.729818,24.783569;46.659321,24.750178

?annotations=distance

http://localhost:5000/table/v1/driving/46.680113,24.715553;46.729818,24.783569;46.659321,24.750178

?annotations=distance

46.680113,24.715553, 46.729818,24.783569,46.659321,24.750178

90.397802,23.716583; 90.409559, 23.731665; 90.390538, 23.750922

http://evrimo.com/bangladesh/table/v1/driving/90.397802,23.716583;90.409559,23.731665;90.390538,23.750922?annotations=distance

<https://evrimo.com/jordan/table/v1/driving/35.726420710830816,32.04796012316255;35.83469427763347,31.98909117233351?annotations=distance>

<https://evrimo.com/singapore/table/v1/driving/103.83325230732939,1.4382353118401097;103.77694737770129,1.3489978103546434?annotations=distance>

3.2211906421980867, 101.67820168570131

3.5227912419994136, 101.78806496302441

**Memory Issue Solution:**

https://programmerah.com/g-internal-compiler-error-killed-program-cc1plus-6453/