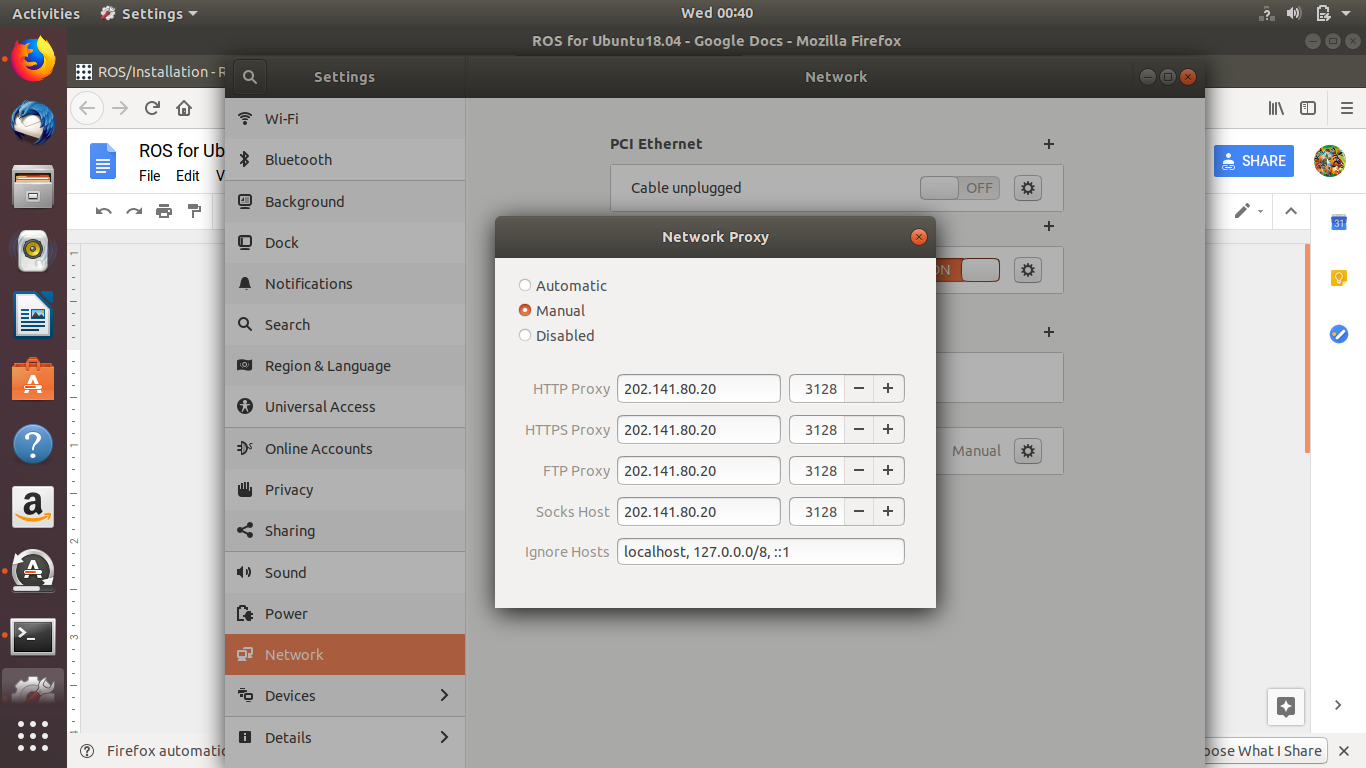
You can either do the whole process on mobile hotspot or switch between proxy and hotspot

Whenever you are asked to enable proxy you have to do it at two places

One is in Network



and the other at /etc/apt/apt.conf (create a new file if it doesn’t exist using

**sudo gedit /etc/apt/apt.conf** and then enter your proxy settings there in the following format

Acquire::http::Proxy "http://username:password@server:port";

Acquire::https::Proxy "https://username:password@server:port";

Acquire::ftp::Proxy "ftp://username:password@server:port";

Acquire::socks::Proxy "socks://username:password@server:port";

and then save and close the file.

Whenever you are asked to turn off proxy you have to undo these two things, so it’s a good idea to save the conf file somewhere for quick copy pasting.

For undoing the settings (**sudo mv /etc/apt/apt.conf ~/**)

For setting it again (**sudo mv ~/apt.conf /etc/apt/**) Whenever proxy is asked to turn off you are assumed to connect to your mobile hotspot and when on, to iitg network.

Turn off the proxy and run the following commands

**sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb\_release -sc) main" > /etc/apt/sources.list.d/ros-latest.list'**

**sudo apt-key adv --keyserver hkp://ha.pool.sks-keyservers.net:80 --recv-key 421C365BD9FF1F717815A3895523BAEEB01FA116**

Turn on the proxy

**sudo apt update**

**sudo apt upgrade**

**sudo apt install ros-melodic-desktop-full**

**sudo apt install ros-melodic-mavros**

**sudo apt install ros-melodic-mavros-extras**

**sudo apt install python-jinja2 python-numpy python-toml**

Now turn off the proxy

**sudo rosdep init**

**rosdep update**

**echo "source /opt/ros/melodic/setup.bash" >> ~/.bashrc**

Now open a new terminal and enter

**roscore**

to see if the installation of ros was successful

Now close all terminals and then go to this website

https://raw.githubusercontent.com/mavlink/mavros/master/mavros/scripts/install\_geographiclib\_datasets.sh

copy the code from there and paste it inside a new file using

(in a new terminal type)

**gedit geo.sh**

and paste the code there, save and close it.

Now source the file using the following command

**sudo bash geo.sh**

Now turn on the proxy and then in new terminal enter the following commands

**sudo apt install git**

**git config --global http.proxy http://username:password@server:port**

**mkdir src**

**cd src**

**git clone https://github.com/PX4/Firmware.git**

Now turn off the proxy (this is the last time)

In a new terminal

**git config --global --unset http.proxy**

**cd src/Firmware**

**make px4\_sitl\_default gazebo** (this may launch the gazebo simulator, once it does check that there is a drone in the middle inside the simulator)

close the simulator and Ctrl+C in the current terminal.

In a new terminal

**mkdir sim\_ws**

**cd sim\_ws**

**mkdir src**

**catkin\_make**

**gedit ~/.bashrc** ( .bashrc is a script that runs every time you open a new terminal, so all the commands that you want to run every time you open a new terminal can be put here)

add the following lines to the end of the file

source ~/sim\_ws/devel/setup.bash

source ~/src/Firmware/Tools/setup\_gazebo.bash ~/src/Firmware ~/src/Firmware/build/px4\_sitl\_default

export ROS\_PACKAGE\_PATH=$ROS\_PACKAGE\_PATH:~/src/Firmware

export ROS\_PACKAGE\_PATH=$ROS\_PACKAGE\_PATH:~/src/Firmware/Tools/sitl\_gazebo

now save the file and close it.

In a new terminal

**cd sim\_ws/src**

**catkin\_create\_pkg takeoff roscpp mavros geometry\_msgs**

Download the file named CMakeLists.txt from here https://drive.google.com/open?id=1wNnNenC8PSGgl6CMISuLPjugr2dmBRR1

and replace the file at ~/sim\_ws/src/takeoff/ with the same name with this one

and then download the file name offb\_node.cpp from the same drive link and paste it inside the src folder which is inside the takeoff folder ( ~/sim\_ws/src/takeoff/src/ )

Now open a new terminal and enter the following commands

**cd sim\_ws**

**catkin\_make**

**source devel/setup.bash**

open another terminal and run

**roslaunch px4 mavros\_posix\_sitl.launch**

In the previous terminal, run

**rosrun takeoff offb\_node**

You should now see the drone takeoff in the simulator. If it does then everything is installed properly.