MIRACLE
SOFTWARE SYSTEMS

Configuring your Raspberry Pi at Home with the Miracle IoT Developer Kit

DIY Tutorial | Miracle Innovation Labs

Aditya Chinni

Lead – Miracle Innovation Labs
Miracle Software Systems, Inc.

February 27th, 2016

Table of Contents

MIRACLE
SOFTWARE SYSTEMS

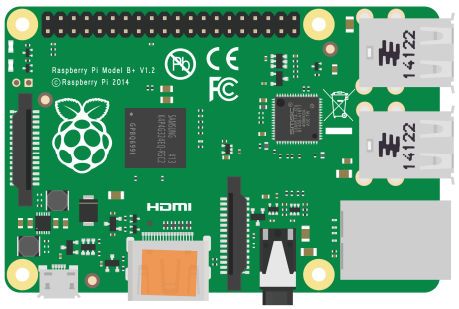
Configuring your Raspberry Pi at Home

Miracle IoT Developer Kit

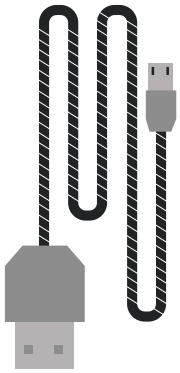
The **Miracle IoT Developer Kit** is pieced together and configured for you to make the best use of a Raspberry Pi and kick start your Internet of Things Journey. The kit comes with the following pieces,

- 1 Raspberry Pi B+ 512 MB RAM
- 1 USB Power Cable
- 1 Micro SD Card with Raspbian Wheezy
- 1 SD Card Adapter
- 1 USB-WIFI Dongle for RPi

The kit comes packaged within our custom made Miracle IoT Developer Kit boxes which are made just for you ☺



Raspberry Pi B+ 512 MB with Case



USB to Micro USB Cable

WIFI USB Dongle

4 GB Micro SD Card(+Adapter) with Raspbian Wheezy Pre-Installed

wlan0 pre-configured for DHCP
elan0 pre-configured for DHCP
800x480 Screen Resolution configured
Samba Server installed
Avahi(ZeroConf) installed
pi@<hostname> with pwd=raspberry
Connection via SSH(Linux and Mac) and Putty (Windows)

Miracle's IoT Developer Kit

Introducing the Raspberry Pi

Raspberry PI is about the size of a credit card, has a 32-bit ARM processor and uses a Raspbian Wheezy distribution of Linux for its default operating system (OS). It can be programmed with Python or any other language that will compile for ARM v6.

In the **Raspberry Pi B+**, you get an HDMI out, RCA video out, 4 USB ports, an SD card slot, a headphone jack, and an Ethernet port. The board itself has half a Half-a-Gigabyte of RAM and an onboard ARM processor.

Connect and Boot your RPi

The first step is to make sure that you have made all the hardware configurations of the Raspberry Pi correctly. You will need to connect the following items to the RPi.

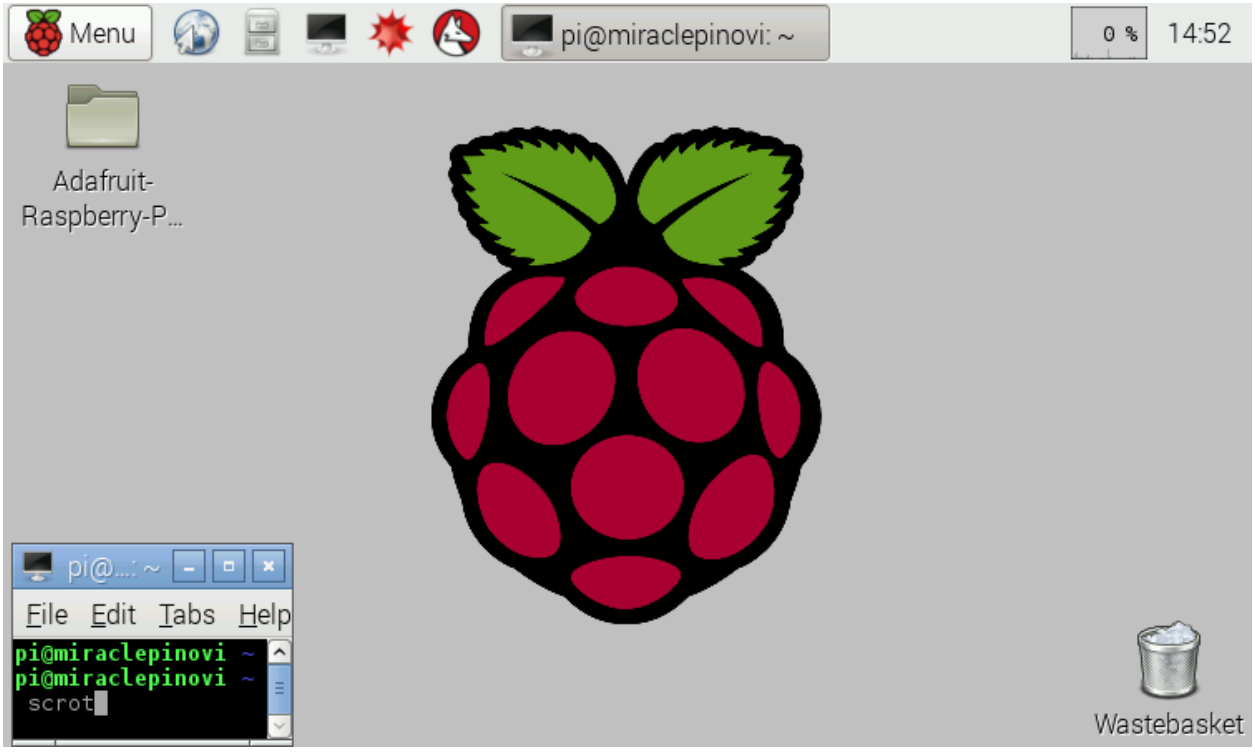
- **USB-WIFI Dongle** (or) **Ethernet Cable**
- **HDMI cable to Monitor**(Optional, if using in headless mode)
- **Keyboard and Mouse**(USB) – (Optional, is using in headless mode)
- **Memory Card(4-8GB) with Raspbian Wheezy**(Included in the Miracle IoT Developer Kit by default)
- **USB Power Source**(As a best practice plug in the RPi to power after all the above have been connected)

Once you plug in your RPi, it will boot up automatically. You will be able to see a red light and a blinking green light. The green light will blink during the boot process, and once the RPi is booted up it will cease.

Note : For the first time you will need to connect to a HDMI monitor to work with your Pi. If you are already able to connect to the Internet with your Pi, then you can access your Pi via SSH in headless mode as well.

The default username for the device is “**pi**” and the password will be “**raspberry**”.

Once you login use the **startx** command to start the UI screen of your RPi. You can also just use the command line interface to work with your PI as well.



Enable Internet for your RPi

Network Set up Steps

Once your RPi is powered and booted we need to set up the Network Configurations of our RPi to connect it to the Internet. Since we do not have connectivity as of now we will need to use it with a HDMI monitor to change the network settings.

Note : By default RPi comes with eth0 set up to DHCP, you can simply plug in the Ethernet cable in to RPi and can ping the hostname of the RPi and you should able to see the ping along with the RPi IP Address. If you see this you can SSH into the device and change the network settings as needed.

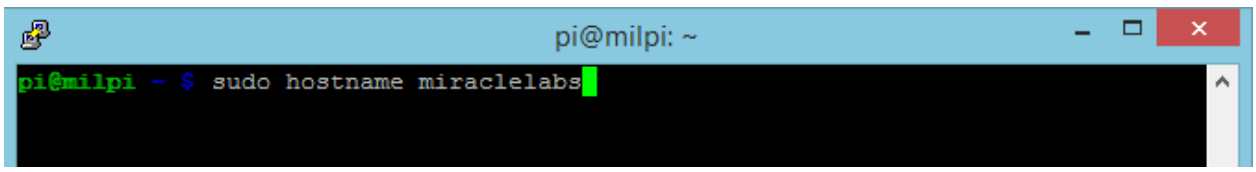
Once you have booted up your RPi use the below commands with the command line (or) within the terminal in the RPi's UI.

Changing your RPi's Hostname

Before giving your RPi a new hostname, you can check the hostname of the device with the **hostname** command.

Now, change the hostname with the following command,

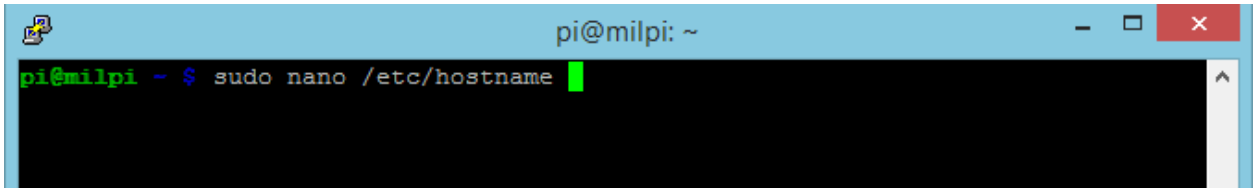
sudo hostname <new hostname>



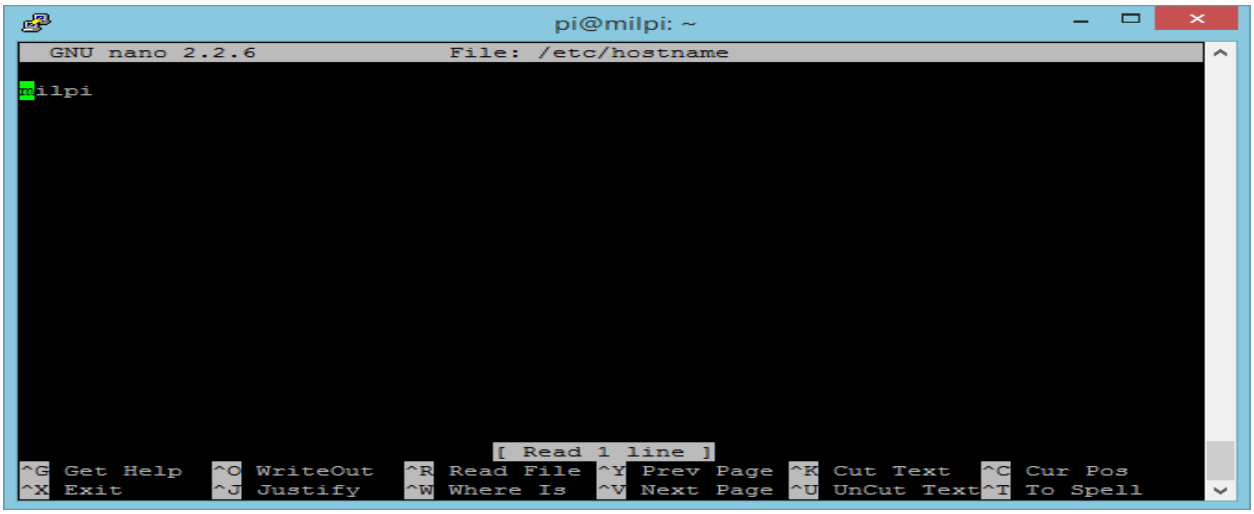
You can see that the host name has been changed with the **hostname** command.

The above command will set the hostname temporarily which will remain till a reboot occur. In order to make it permanent, edit **/etc/hostname** file with **nano** editor using the following command,

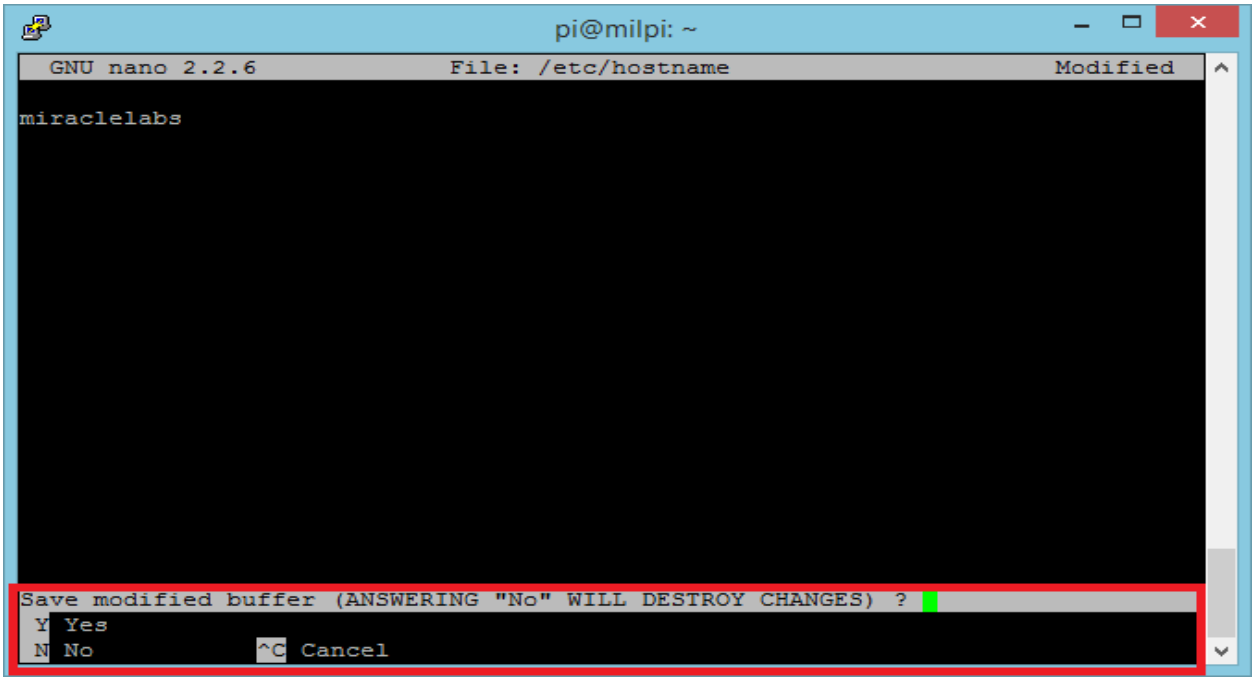
sudo nano /etc/hostname



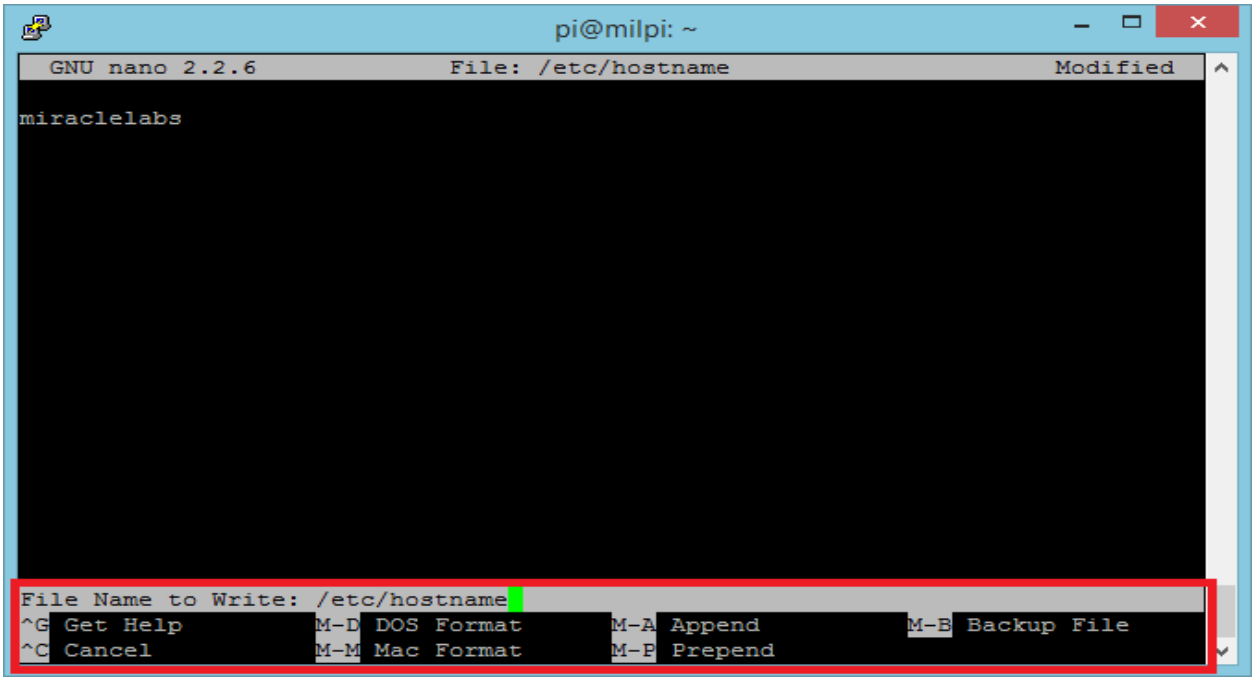
The file will show you the already existing hostname, replace it with the new hostname that you have specified earlier.



After replacing the hostname, press **ctrl+x** to save the file and then It will ask you whether to save the modified buffer or not. In order to save press Y or else N.



Now, it will ask for the filename to save by showing the original file name. If you want to modify the file name just type the new name of the file with which you would like to save or else leave the name as it is and press Enter.



After saving the file, you need to reboot to let the changes take place with the **sudo reboot** command.

Change the Network Configuration

In a network, an IP is a unique number which is used to identify a system. In general, you have two ways of assigning IP.

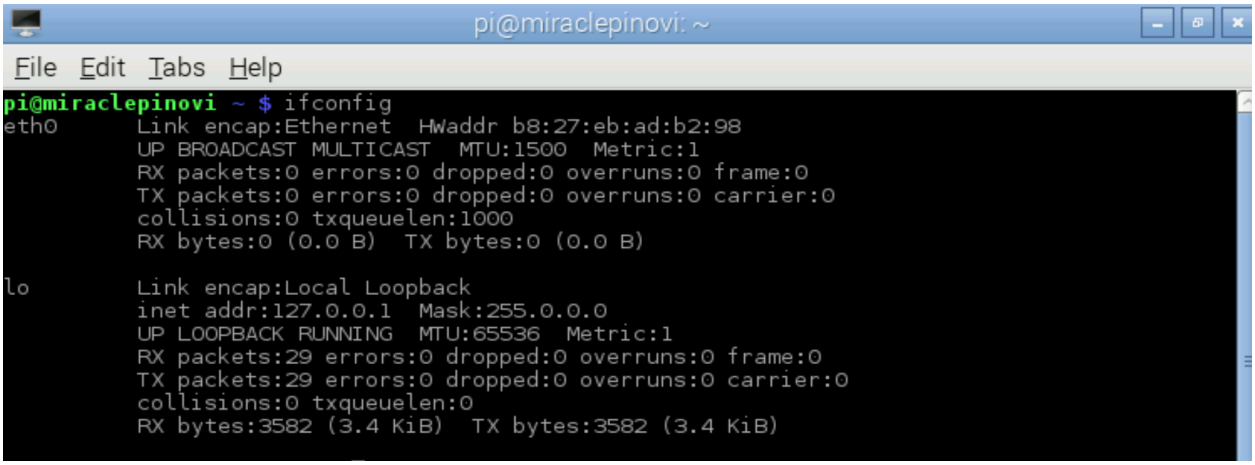
- **Static Addressing**
  - In static IP, you will assign a device with a unique IP which won't change even after reboot and every time the network uses the same IP to identify that device and you can use the same IP to connect to that device from a remote system.
- **Dynamic Addressing**
  - In dynamic addressing, a device will be assigned to a IP among all the available IP's in a network, which may change on reboot. So, for a system with dynamic IP you can't use the same IP every time to connect it from the remote system.

**By default your RPi can be used in DHCP, but you can set to static as well*

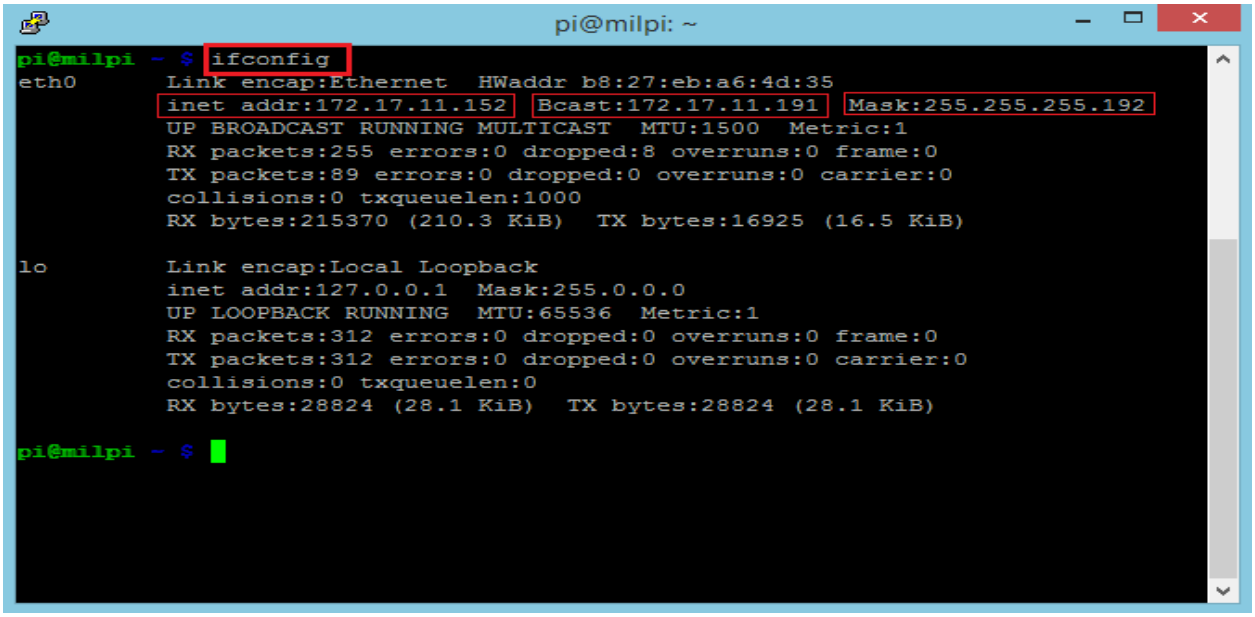
Assign a Static IP Address

In order to assign a static IP to Raspberry Pi, you need to first gather the details of the **address, netmask, network, broadcast, gateway** and **DNS** of your current network. In order to get the address, broadcast and netmask simply enter the **ifconfig** command in the command prompt.

But since initially we haven't configured or set up any IP address for the RPi and you can see at eth0 there is no IP Address resolved.



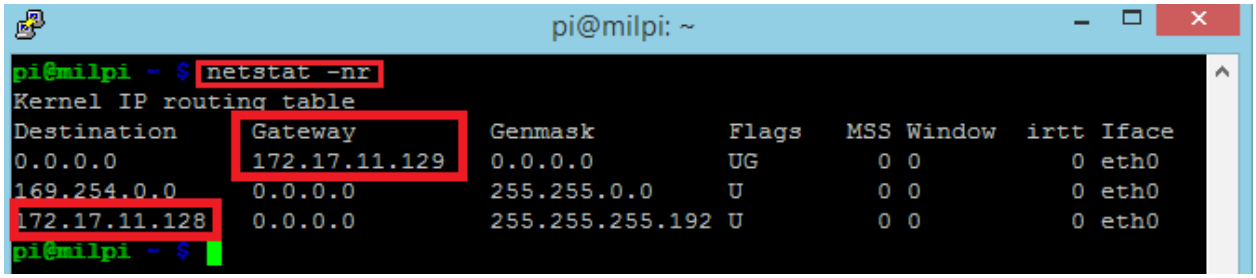
Now try to plug in the Ethernet cable into the RPi and reboot it with the command "**sudo reboot**". Once the reboot is done login back to the RPi as mentioned above and use the **ifconifg** command again.



You can now see that your RPi has been assigned a dynamic IP Address as follows,

Address = 172.17.11.152
Broadcast = 172.17.11.191
Netmask = 55.255.255.192

In order to get the gateway and network details, you can use the **netstat –nr** command.

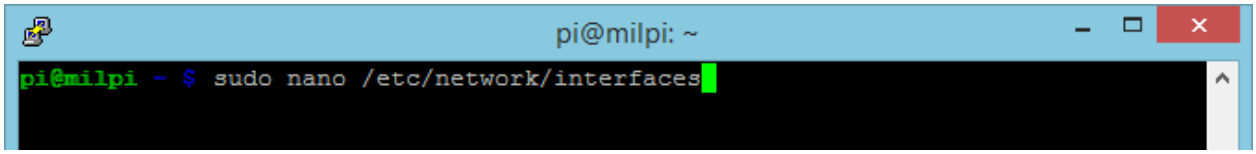


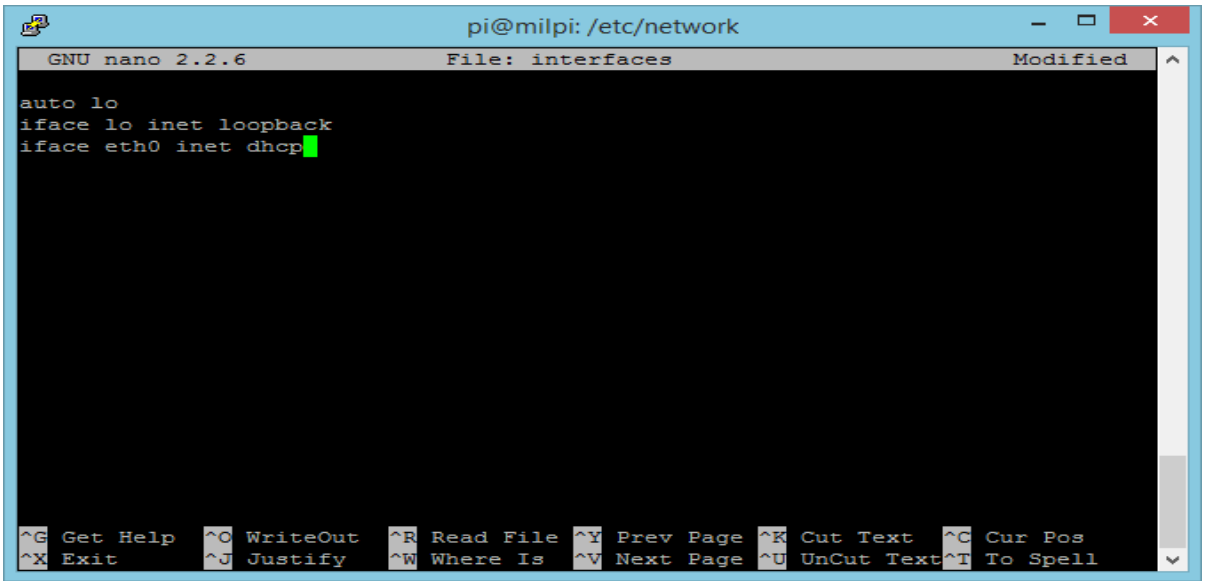
From the above you can observe the network and gateway as,

Network = 172.17.11.128 **Gateway** = 172.17.11.129

You can now set your RPi's static IP Address by editing **/etc/network/interfaces** file using **nano** editor using the following command,

sudo nano /etc/network/interfaces





Change the file by adding the following details that you gathered and press **ctrl+x** to close the file.

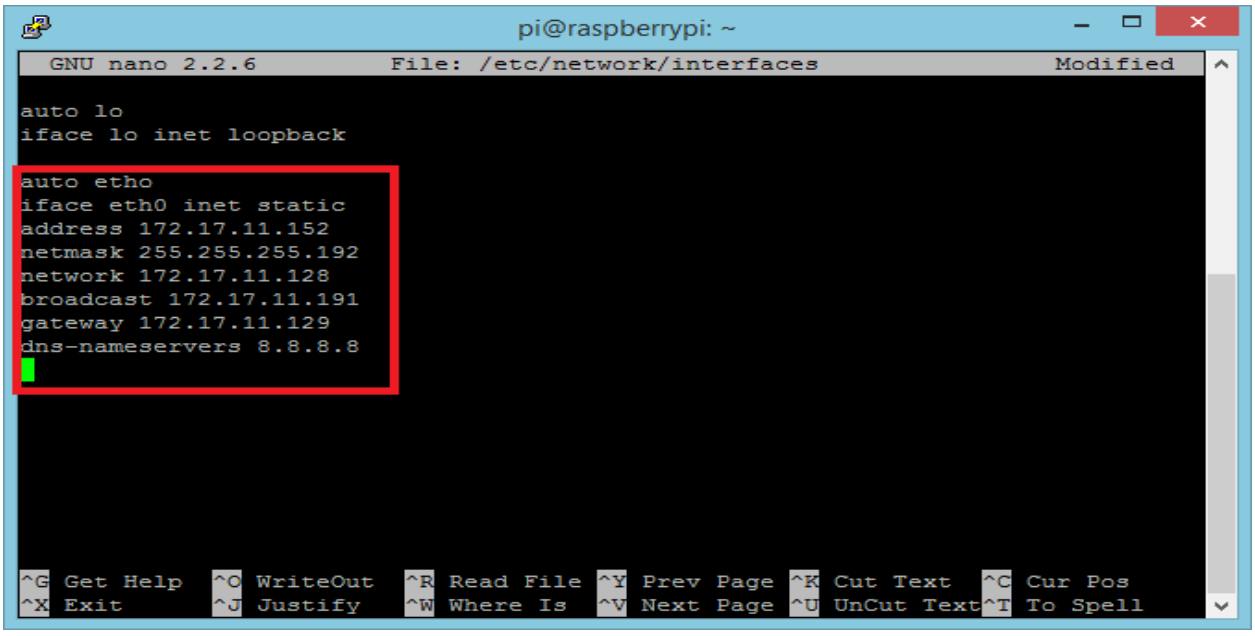
Remove

iface eth0 inet dhcp

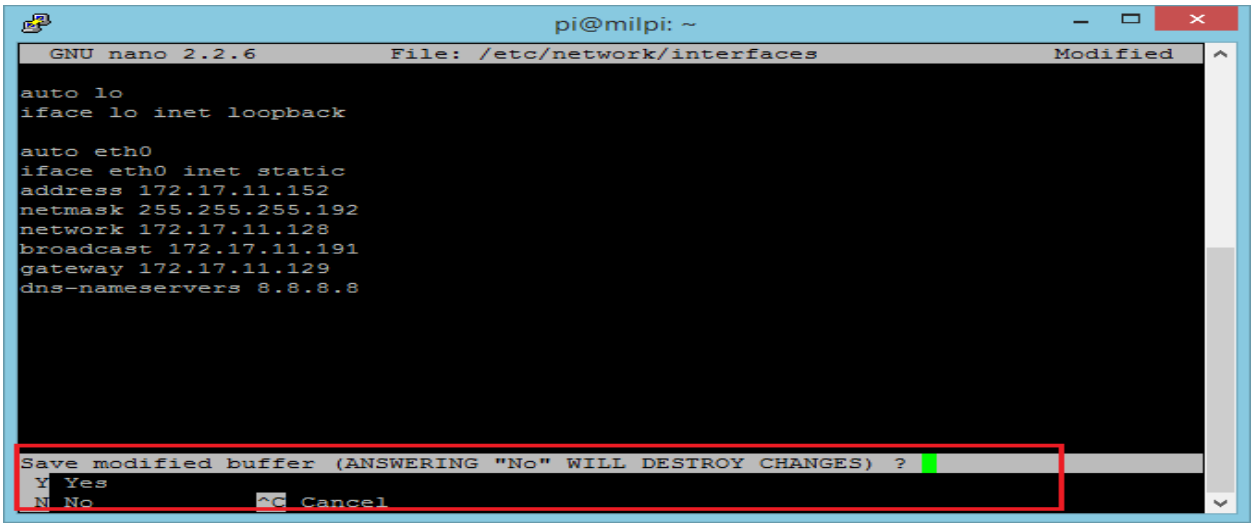
Add

auto etho
iface eth0 inet static
address 172.17.11.152
netmask 255.255.255.192
network 172.17.11.128
broadcast 172.17.11.191
gateway 172.17.11.129
dns-nameservers 8.8.8.8

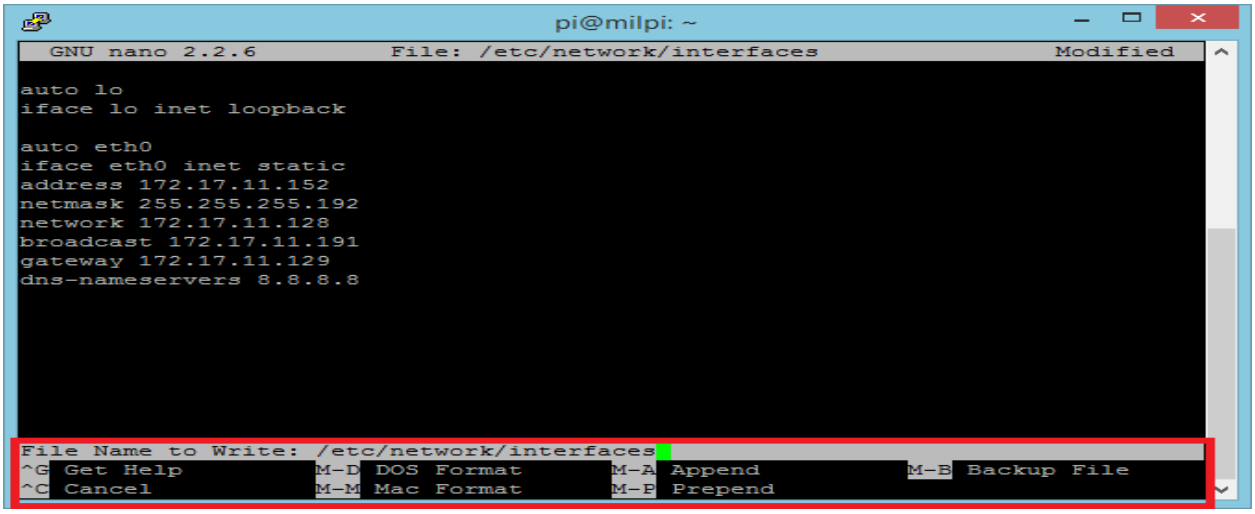
Note : All the details should be filled from the details that you have collected and **dns-nameservers** may change depending on the network provider.



It will ask you whether to save modified buffer or not. In order to save press Y.

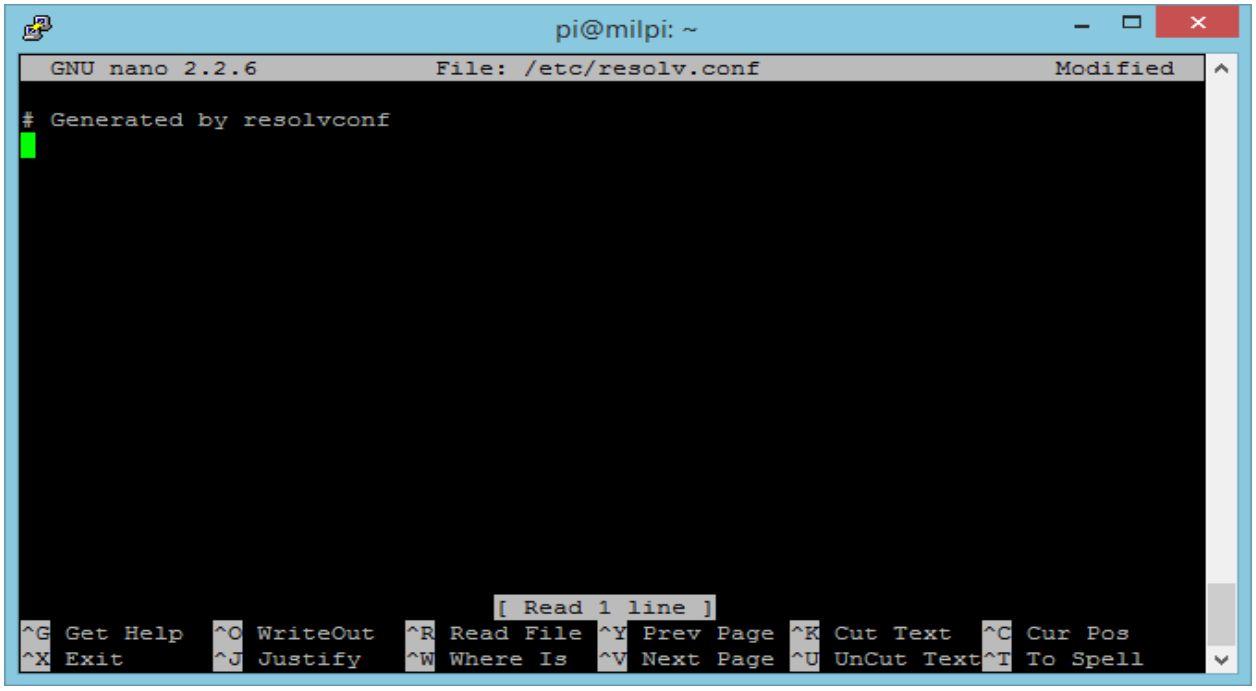


Now, it will ask for the filename to save by showing the original file name. If you want to modify the file name just type the new name of the file with which you would like to save or else leave the name as it is and press Enter.

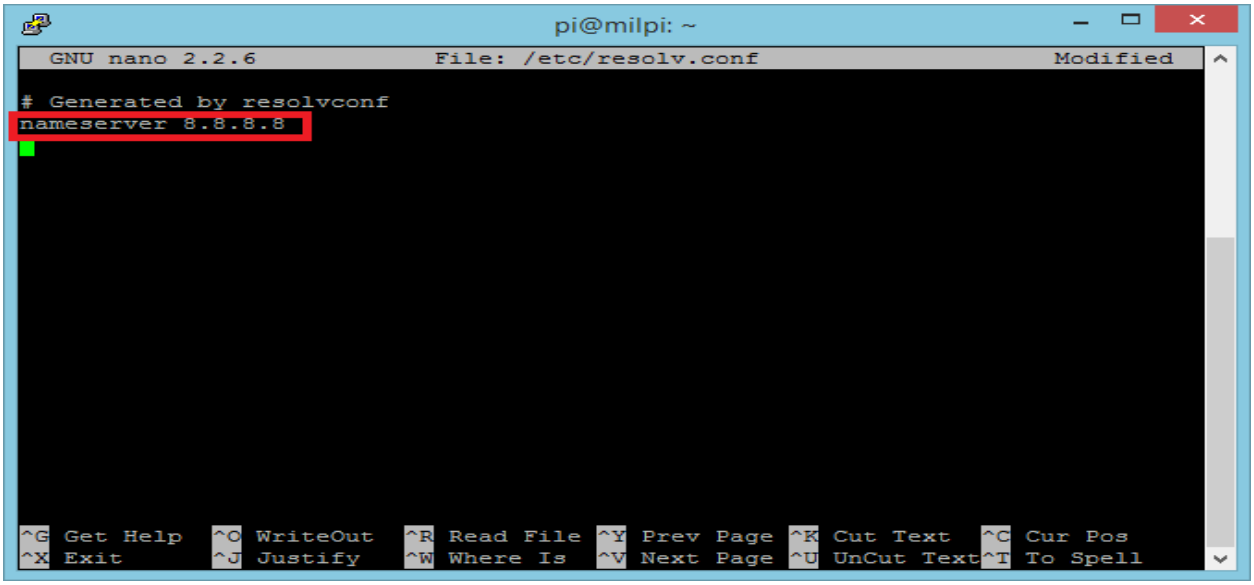


You need to specify the DNS resolving name by editing **/etc/resolv.conf** with nano editor using the following command.

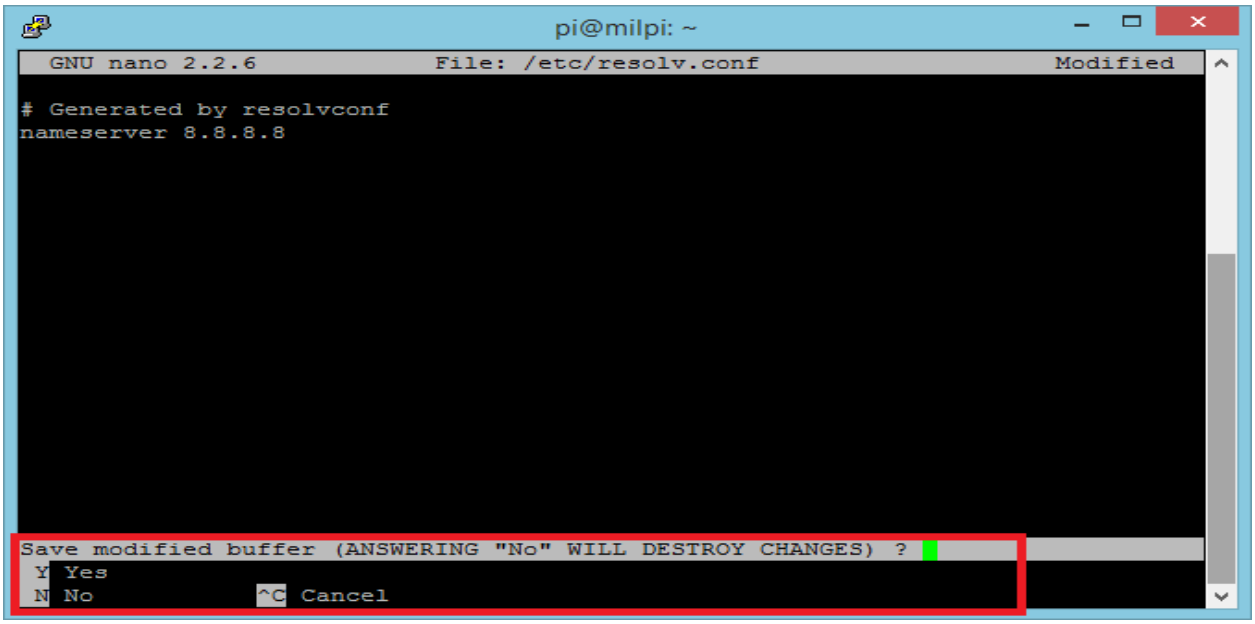
sudo nano /etc/resolv.conf



Enter the DNS details as **nameserver 8.8.8.8**



Now, press **ctrl+x** to save the file. Then, it will ask you whether to save modified buffer or not. In order to save press Y.

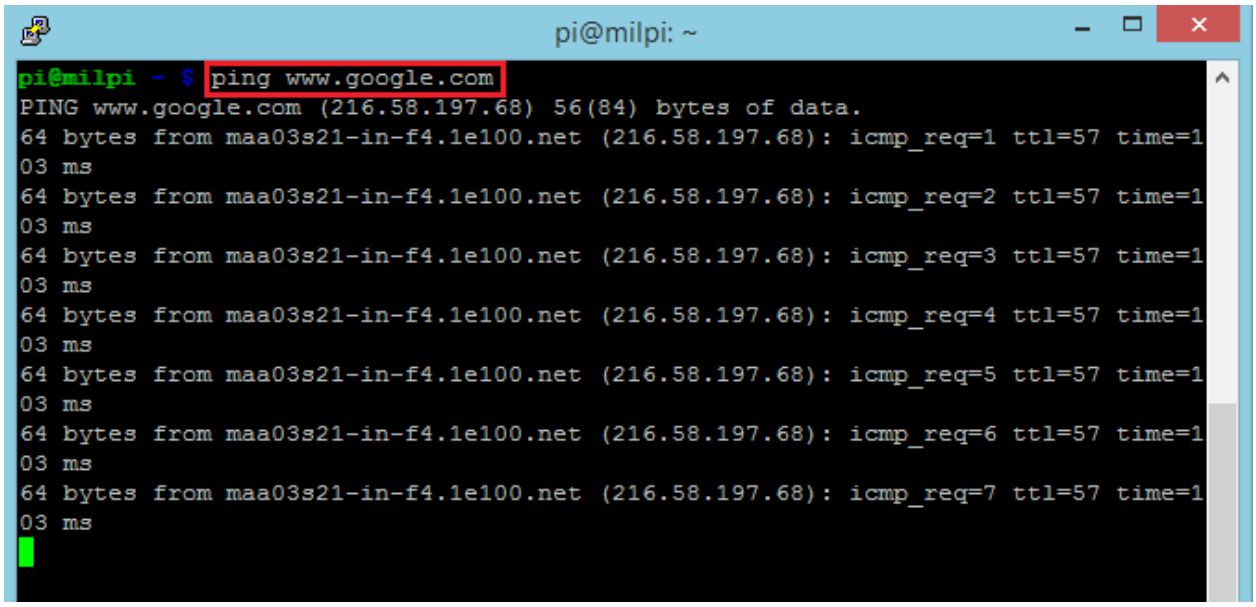


It will ask for the filename to save by showing the original file name, if you want to modify the file name just type the new name of the file with which you would like to save or else leave the name as it is and press Enter.

You need to reboot the RPi for the changes to take place by using the **sudo reboot** command.

Verify whether the system is able to access the network by pinging to the Google server using the following command

ping www.google.com

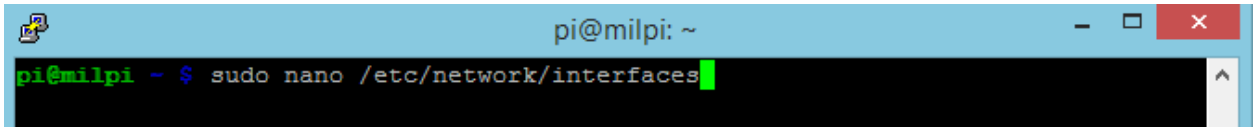


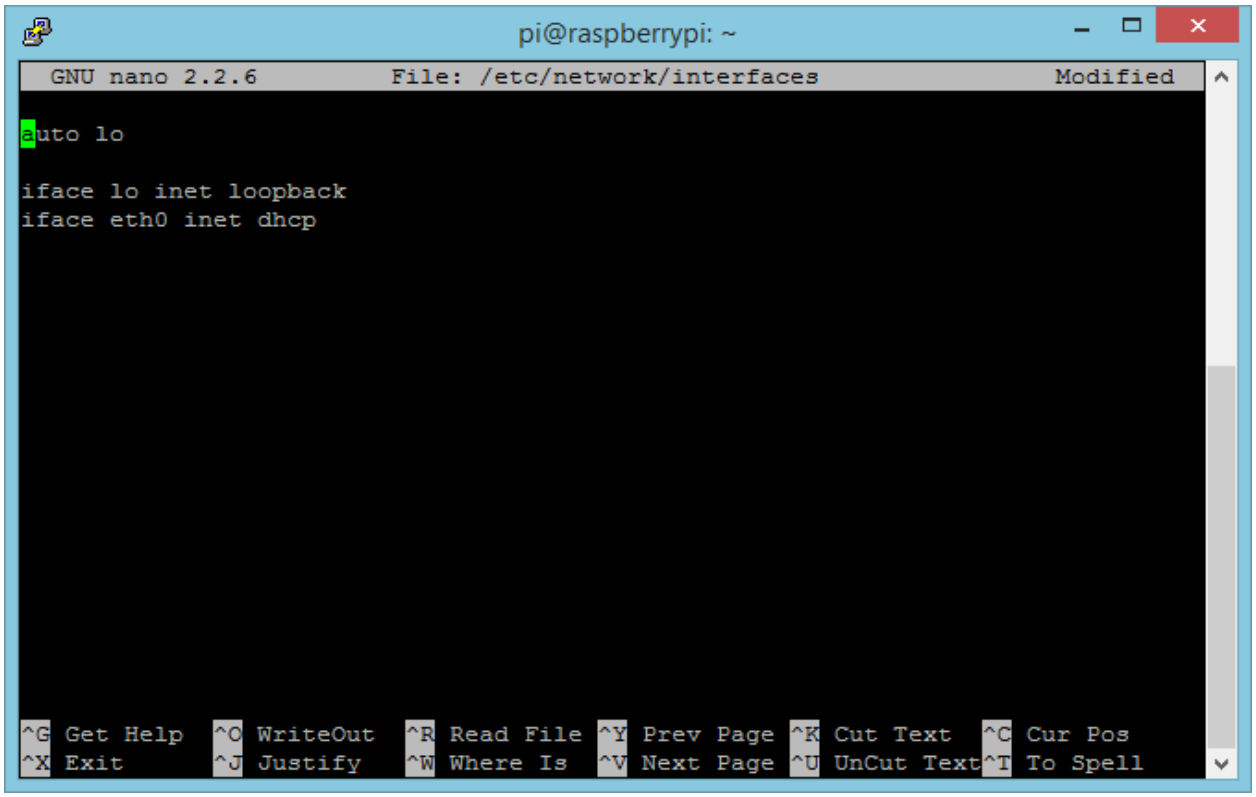
Press **ctrl+C** to stop pinging.

Configure WIFI(wlan0) for your RPi

In order to configure wifi and use the wireless based Raspberry Pi, you need to have the wifi adapter connected to the Raspberry Pi device. In order to set wifi configuration, you need to edit **/etc/network/interfaces** file using nano editor.

sudo nano /etc/network/interfaces

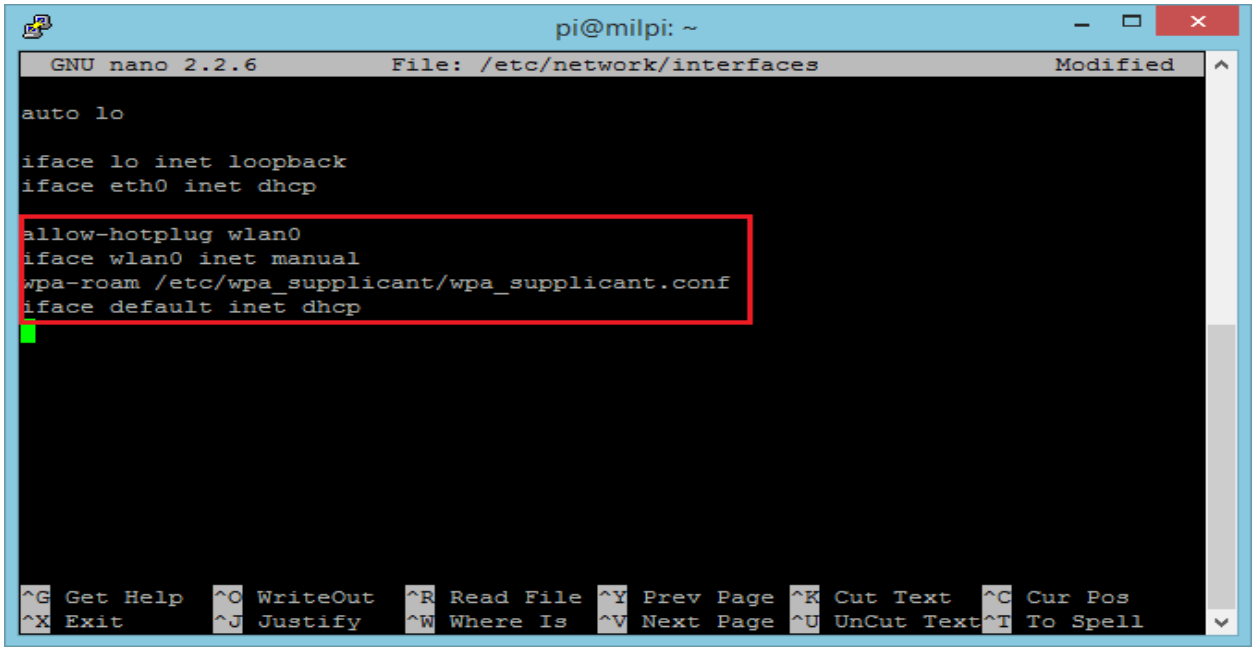




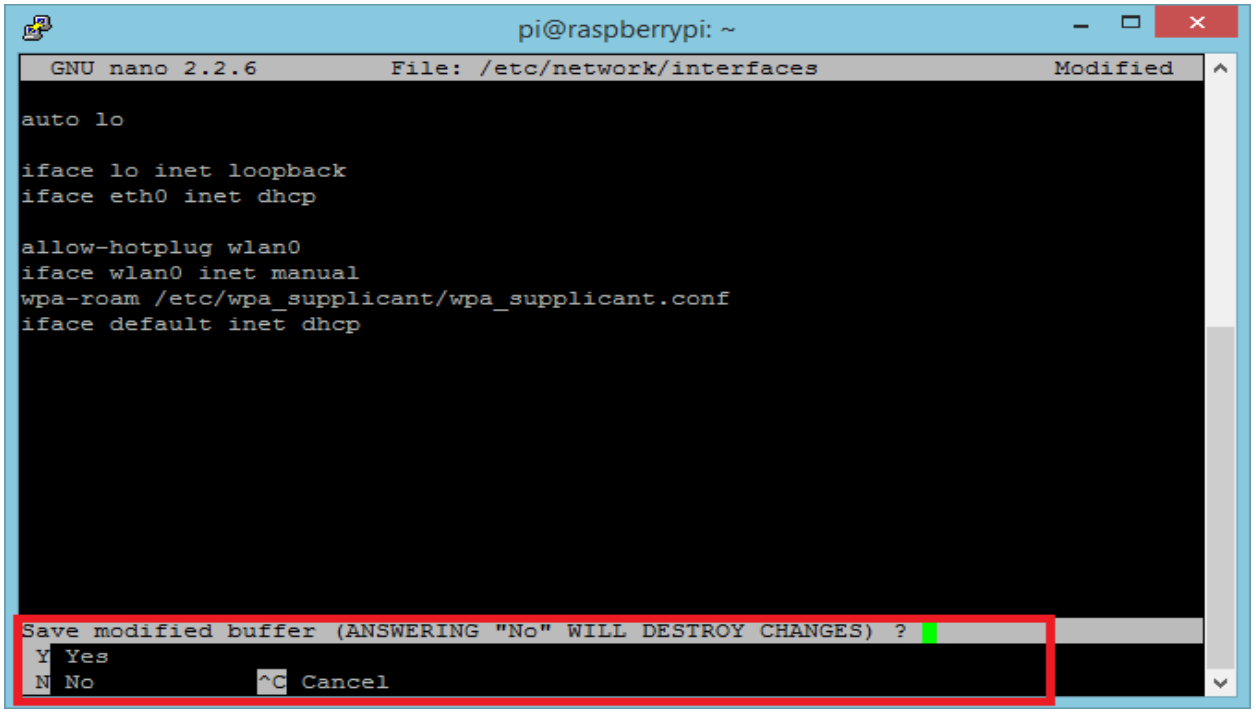
Change the content of the file as specified below and press **ctrl+x**. Add the following content at the bottom of the existing data.

allow-hotplug wlan0

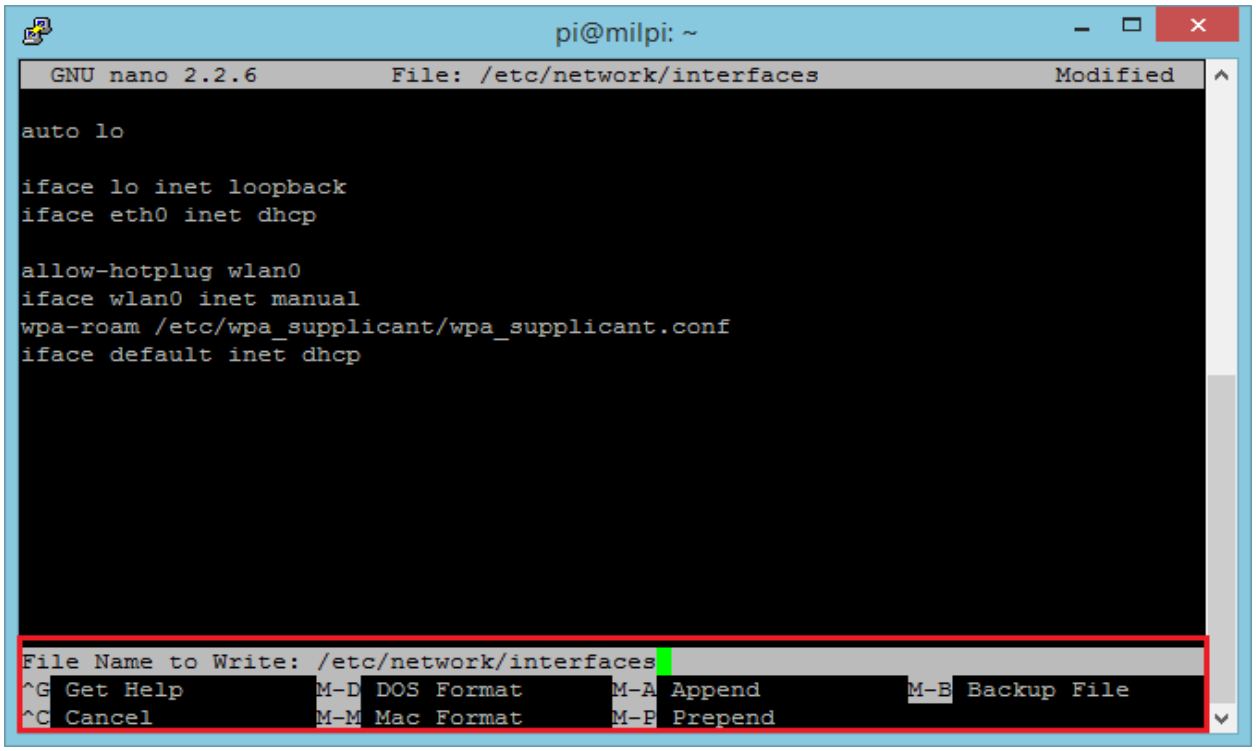
iface wlan0 inet manual
wpa-roam /etc/wpa_supplicant/wpa_supplicant.conf
iface default inet dhcp



It will ask you whether to save modified buffer or not. In order to save press Y.



It will ask for the filename to save by showing the original file name. If you want to modify the file name just type the new name of the file with which you would like to save or else leave the name as it is and press Enter.



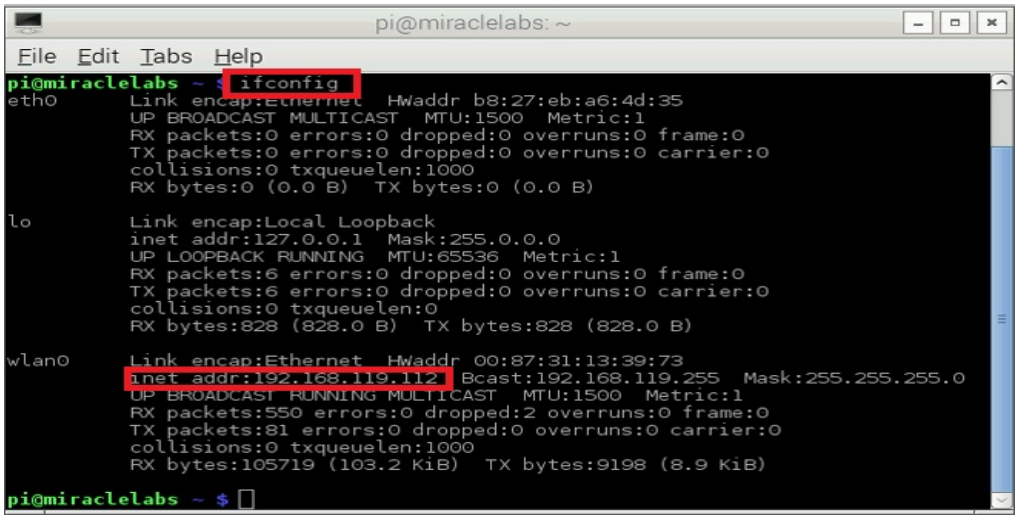
Now we need to Configure WiFi **"wpa-supplicant.config"** file.

sudo nano /etc/wpa-supplicant/wpa-supplicant.config

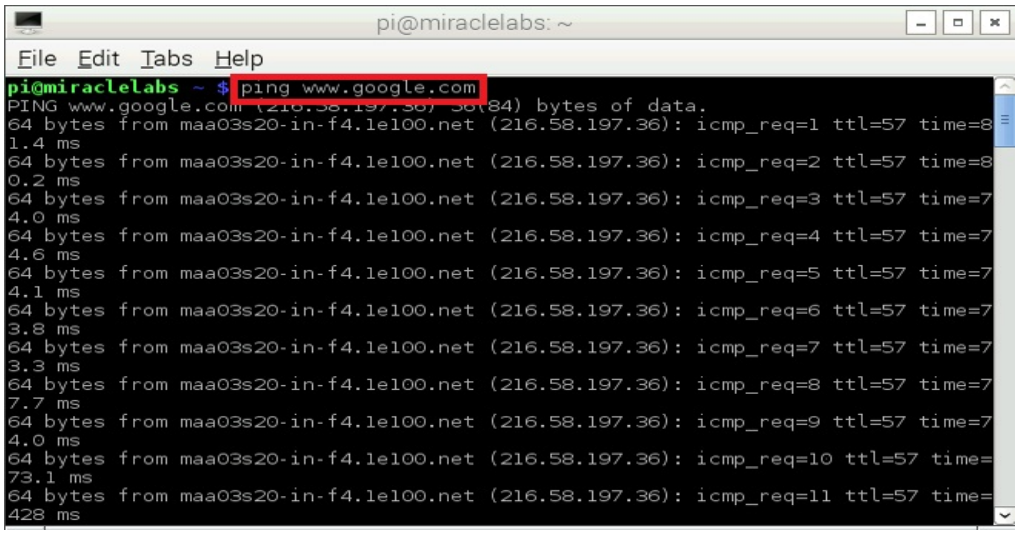
Add the below details to the file.

```
network={
        ssid = "Your SSID Here"
        proto = RSN
        key_mgmt = WPA-PSK
        pairwise = CCMP TKIP
        group = CCMP TKIP
        psk = "Your Password Here"
        }
```


Once this is done click on **“ctrl+x”** and then click on **“y”** and then **“enter”** to save your file. Reboot your RPi and then check the network with **ifconfig.**



Ping to any of the servers to ensure that you are able to access the Internet as shown below.



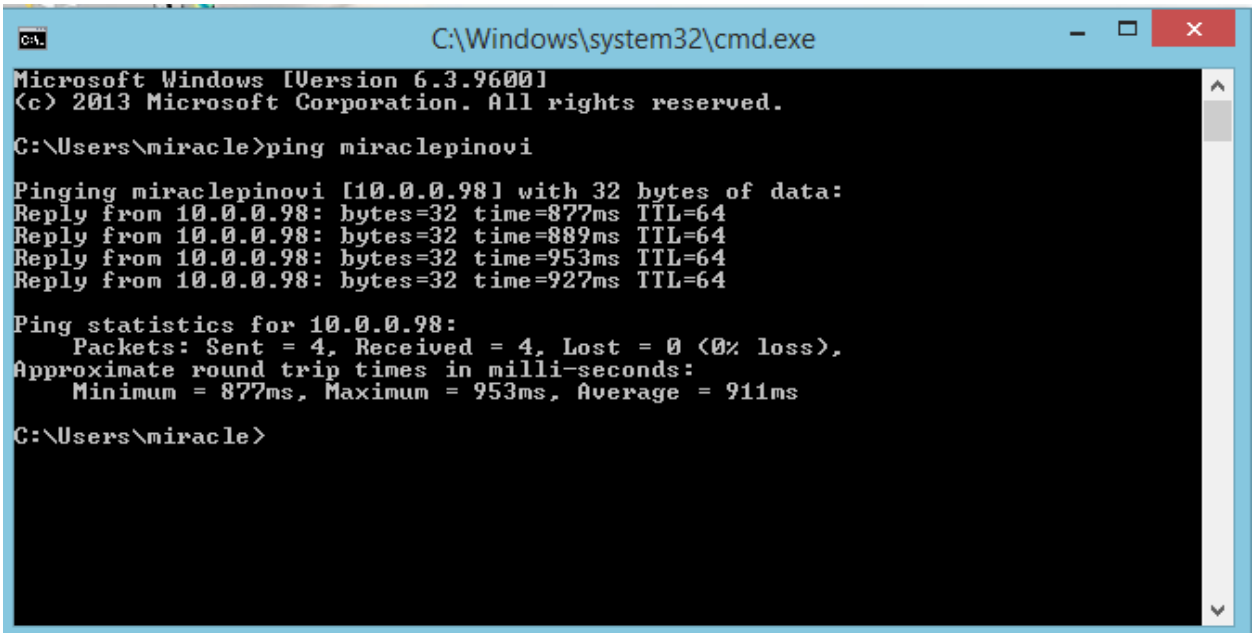
Use **ctrl+c** to exit out of the ping command.

Connecting to your RPi

Open Command Prompt and enter the below command where “miraclepinovi” is the hostname of my RPi and hit enter. You can also check with your static IP address based on which network option you chose above.

ping miracelpinovi (or) ping <ip address>

**For Mac and Linux you might need to use <hostname>.local to ping your device*



Connecting to your Pi using SSH

In order to work with Raspberry Pi from a remote system, you need to have ssh service enabled in our system. In windows, to gain access to ssh service, you need software called **Putty,** which provides us a console to execute our commands. **Before that, you need to know raspberry Pi's IP (or) hostname.**

Install SSH in Linux Machine

In order to connect it from Linux systems, you should have ssh client installed in your system. The installation differs from each and every flavour of Linux. So,

execute the following command in the command prompt based on your operating system.

Ubuntu : sudo apt-get install -y -f openssh*
Redhat (or) Centos : yum install -y openssh*

Now, you need to restart the ssh service using the following command

sudo service ssh restart

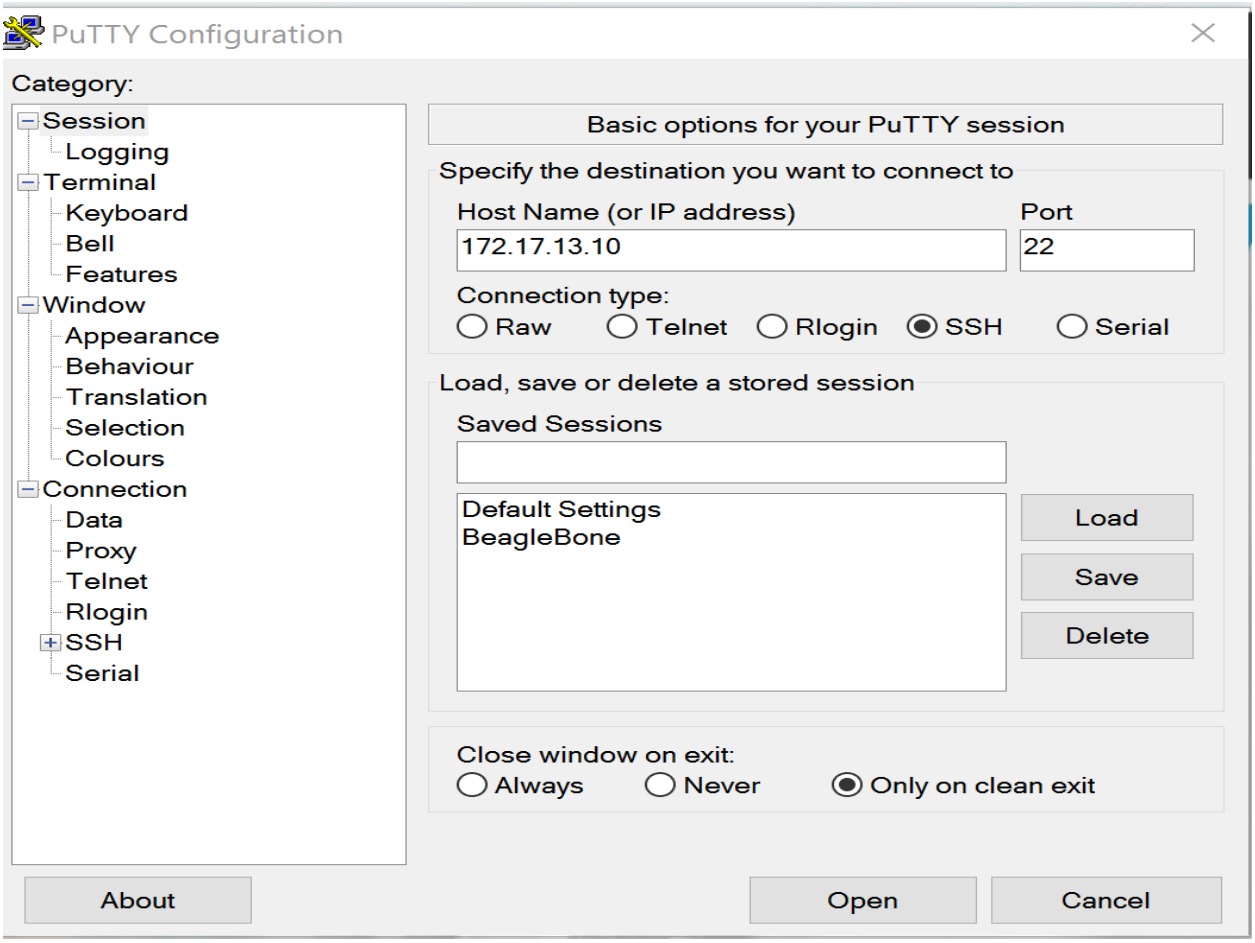
Get Putty in Windows Machine

Download putty from the below link,

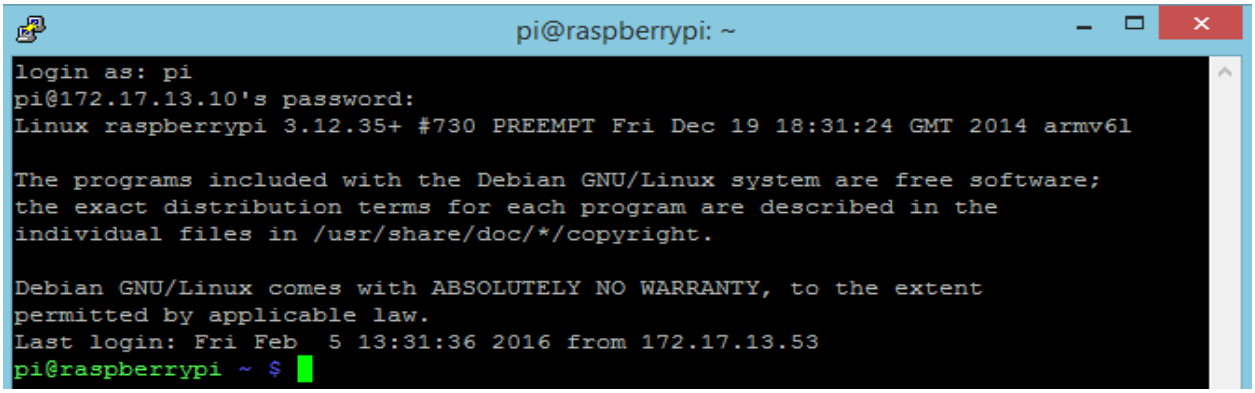
http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html

Download **putty.exe** of development snapshot version from the above link as shown below.

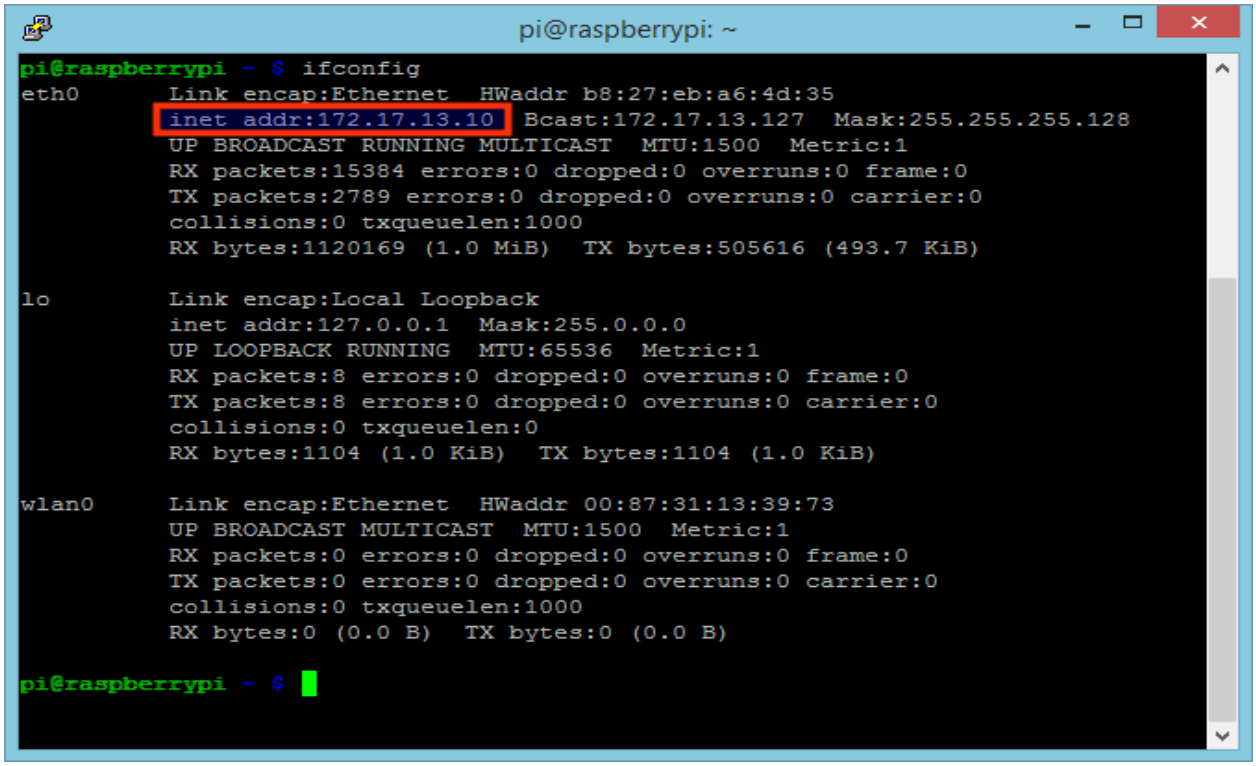
After downloading it successfully, double click on the downloaded **putty.exe** file. Then, it will show a window where you need to specify the IP Address (or) hostname of the RPi that you configured earlier**.** To ssh the remote system, click on open button which is highlighted in the below image.



After clicking on the open button, enter the **login name** as **pi** and **password** as **raspberry**.



To ensure that you have logged into the correct system, use the **ipconfig** command to know the network details of the device.



SSH from Mac OS X

Here is the command for the Mac Users to do SSH in to RPi.

ssh pi@<Hostname of you RPi>

For example **ssh pi@miraclepinovi** and hit enter it will ask you password. By this time you should be more familiar with password of the RPi, which is **raspberry**.