Raspberry Pi: Establishing Network connection and SSH connection

e-Yantra Team

June 15, 2015

Contents

1   Objective

In this tutorial we will be learning how to establish a network connection in an R-Pi and how to connect a PC to a Raspberry Pi using SSH connection(i.e. remote control).

2   Prerequisites

One should be aware of :

- Basic terminal commands.

- The ssid and password of a wireless network to which an R-Pi has to be connected.

3   Hardware Requirement

1. Raspberry Pi (I will be using Version 2 Model B)

2. Monitor,HDMI cable,Keyboard,Mouse (This an optional mode of connecting an R-Pi.In order to learn about different ways to connect an R-Pi kindly refer the previous tutorial)

3. Wireless adapter

4. Power adapter

5. PC(either Windows or Linux)

4   Software Requirement

MobaXterm(for Windows user)

5   Theory and Description

Secure Shell is a program to log into another computer over a network, to execute commands in a remote machine, and to move files from one machine to another. It provides strong authentication and secure communications over insecure channels. It is a replacement for rlogin, rsh, rcp, and rdist. SSH protects a network from attacks such as IP spoofing, IP source routing, and DNS spoofing. An attacker who has managed to take over a network can only force ssh to disconnect. He or she cannot play back the traffic or hijack the connection when encryption is enabled. When using ssh's slogin (instead of rlogin) the entire login session, including transmission of password, is encrypted; therefore it is almost impossible for an outsider to collect passwords. [2] **Applications of SSH or OpenSSH protocol**

Figure 1: [3]

- For login to a shell on a remote host (replacing Telnet and rlogin)

- For executing a single command on a remote host (replacing rsh)

- For setting up automatic (passwordless) login to a remote server (for example, using OpenSSH)

- Secure file transfer

- In combination with rsync to back up, copy and mirror files efficiently and securely

- For forwarding or tunneling a port (not to be confused with a VPN, which routes packets between different networks, or bridges two broadcast domains into one).

- For using as a full-fledged encrypted VPN. Note that only OpenSSH server and client supports this feature.

- For forwarding X from a remote host (possible through multiple intermediate hosts)

- For browsing the web through an encrypted proxy connection with SSH clients that support the SOCKS protocol.

- For securely mounting a directory on a remote server as a filesystem on a local computer using SSHFS.

- For automated remote monitoring and management of servers through one or more of the mechanisms discussed above.

- For development on a mobile or embedded device that supports SSH.[1]

MobaXterm

MobaXterm is a tool used for remote computing. In a single Windows application, it provides loads of functions that are tailored for programmers, webmasters, IT administrators and pretty much all users who need to handle their remote jobs in a more simple fashion.It provides all the important remote network tools (SSH, X11, RDP, VNC, FTP, MOSH, ...) and Unix commands (bash, ls, cat, sed, grep, awk, rsync, ...) to Windows desktop, in a single portable exe file which works out of the box.

There are many advantages of having an All-In-One network application for your remote tasks, e.g. when you use SSH to connect to a remote server, a graphical SFTP browser will automatically pop up in order to directly edit your remote files. Your remote applications will also display seamlessly on your Windows desktop using the embedded X server.[4]

6   Experiments

6.1   Establishing a network connection

Before we can start using a Raspberry Pi remotely we need to configure its network settings.There are two ways to configure network setting:

- One is using DHCP server (wireless connection)

- The other one is by assigning a static IP

6.1.1   Network settings configuration using DHCP Server(wireless settings)

1. Insert the SD card (with Raspbian OS already written) into the micro SD slot in an R-Pi.

2. Connect the wireless adapter, keyboard , mouse and monitor (using a HDMI cable) to the Raspberry Pi.

3. Power on the board and monitor.You will notice a set of code running on the monitor.

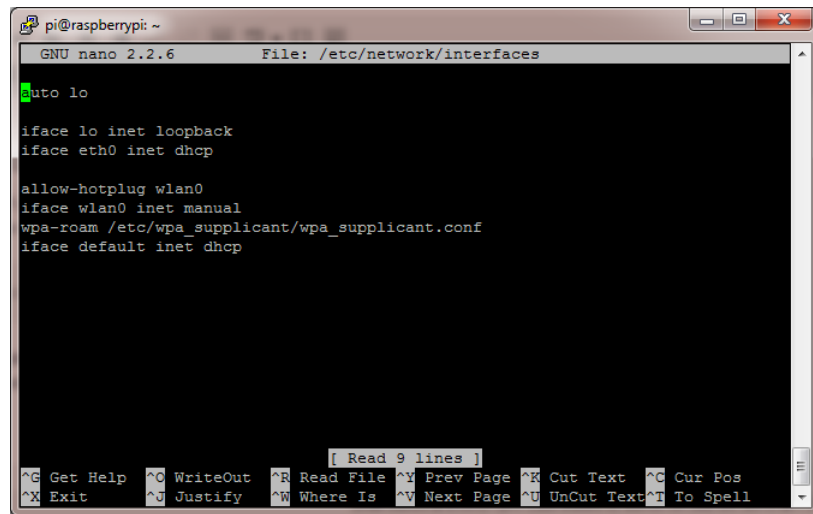4. Enter the set user name and password and then type *startx* command to launch the GUI.
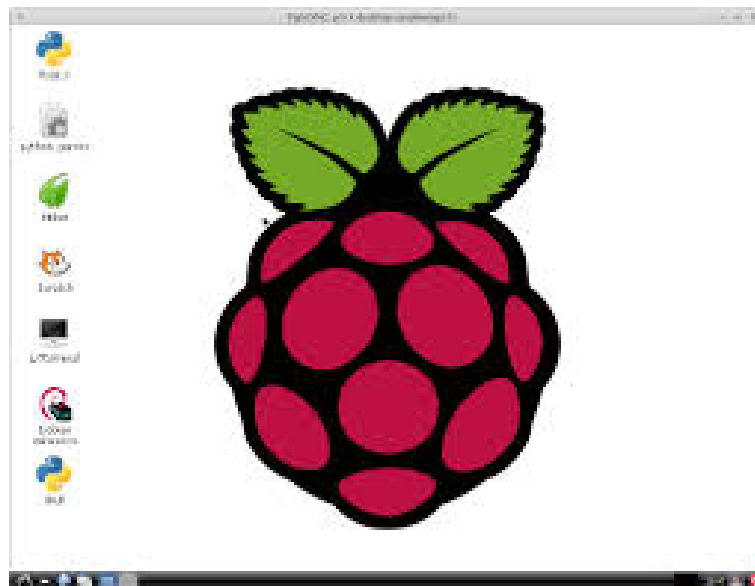
5. Click on the icon at the bottom-left on the screen. This is the start icon.

6. Select LXTerminal option to open a terminal window
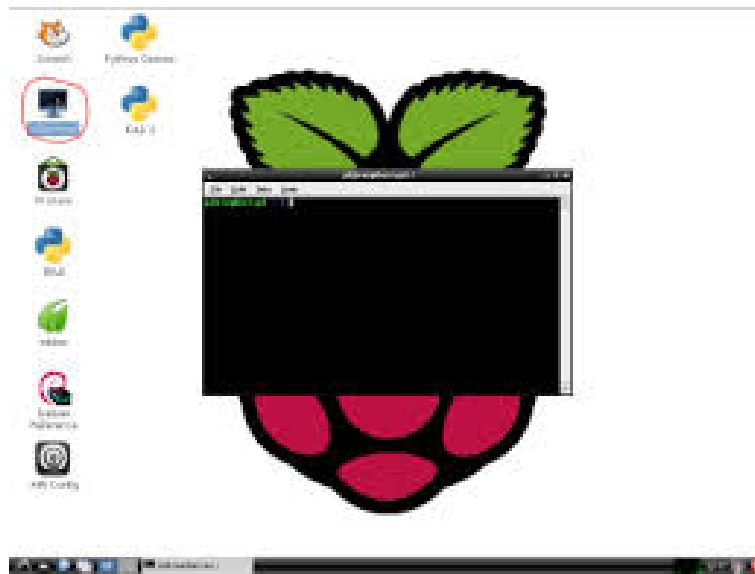
7. Some of the previous versions of Raspbian OS do not support wireless module. To know about the version you are using type *uname -a*. To upgrade to the latest one type "sudo apt-get upgrade".

8. Connect your wireless adapter . To check if its connected type *lsusb*(We used D-link adapter which is highlighted in yellow)
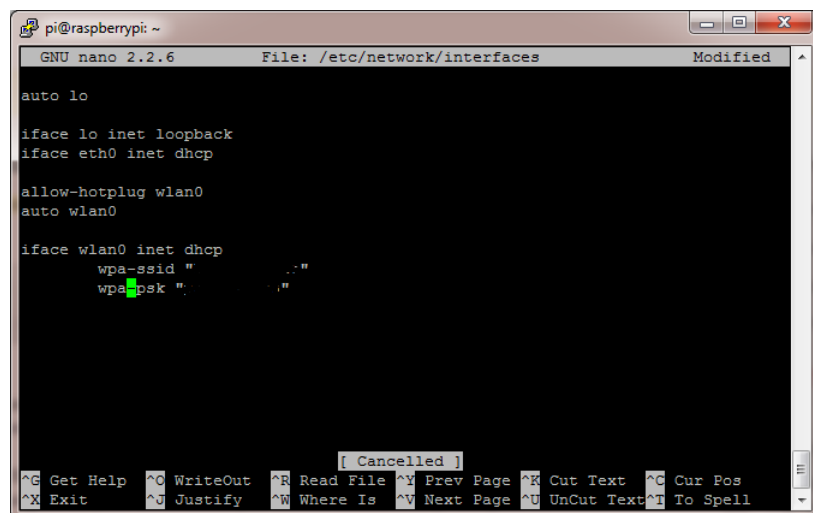

9. Now type *sudo nano /etc/network/interfaces* and press enter. You will see the following window:

10. Change the code by adding the following lines:

11. Add in the SSID(username) and Password of your wifi. Then for changes to take effect type *sudo /etc/init.d/networking restart*

12. Type *ifconfig* to obtain the IP address of R-Pi etc. It will be under wlan0(written as inet address).

6.1.2   Network settings using static IP method

The router normally distributes the dynamic IP addresses but it isnt guaranteed that it will assign the same IP address every time. This can cause problems if you are trying to connect to your Raspberry Pi remotely. And hence we can assign a static IP. In order to do so follow these steps:

1. Open the LXTerminal and type the following command
   cat /etc/network/interfaces

2. Before you make changes to the document you should be aware of your current IP address, the broadcast IP and the Mask Use the command *ifconfig* to retrieve this information To find your gateway address type the following command: *sudo route nee*

3. In order to edit the interfaces file type *sudo nano /etc/network/interfaces*

4. Remove the following line *iface eth0 inet dhcp* and add the following:
   iface eth0 inet static
   address 192.168.0.x
   netmask 255.255.255.0
   network 192.168.0.0
   broadcast 192.168.0.255
   gateway 192.168.0.y

5. Save the file using Ctrl+X, Y.

6. For changes to take effect type *sudo /etc/init.d/networking restart* and reboot the system.
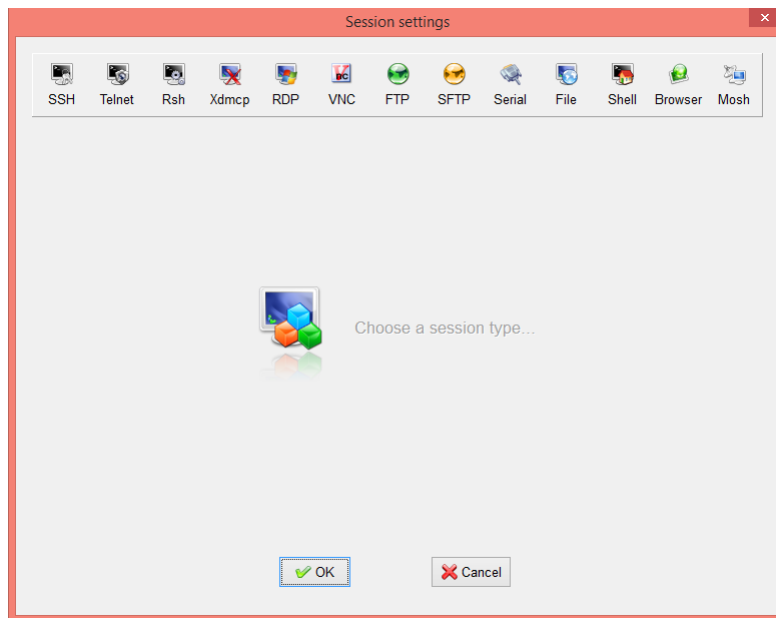
6.2   Establishing an SSH connection

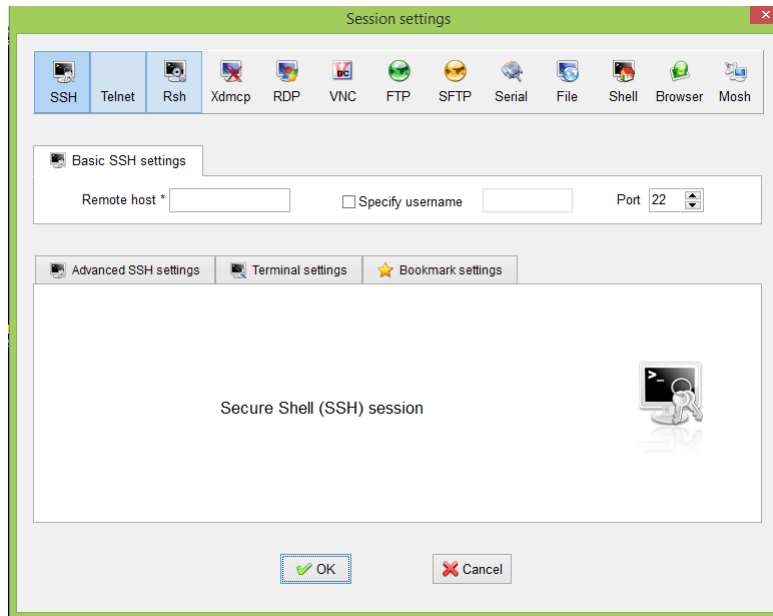To start using a Raspberry Pi remotely follow these steps:

1. A windows user should download MobaXterm (latest version) using
   the following link: `http://mobaxterm.mobatek.net/`

2. Run the .exe file and open the application.
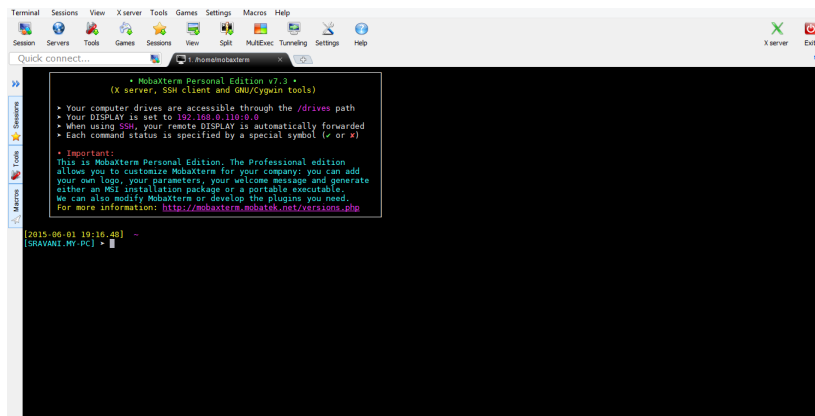
3. Then select the session option on the toolbar.

4. Click on the SSH option. A settings window opens as shown

5. Enter the IP address of the R-pi in the Remote host field and in the 'Advanced SSH settings' option configure the remote environment either as 'Interactive shell (terminal based)' or as an 'LXDE desktop(GUI based)'and click OK

6. If you are using the Interactive shell environment login to the R-Pi using the following command: ssh -X pi@192.168.0.4 (IP address of pi) and then enter the login id and password to start using the Pi remotely.

7. If you are using the LXDE desktop i.e. the GUI environment then use LX terminal icon to start programming the Pi.
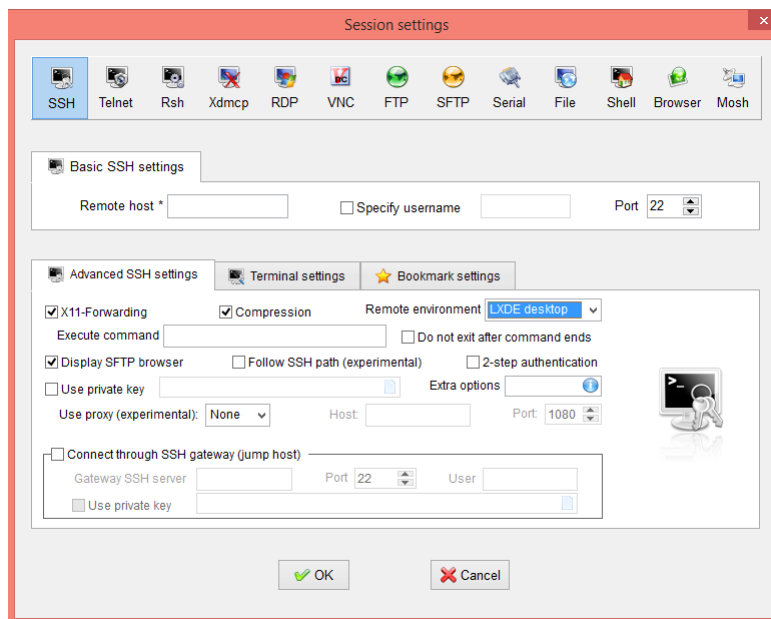
8. Also the established session is saved. So the next time you want to access directly click on the R-Pi's address mentioned in the saved session option to start using the Pi remotely.
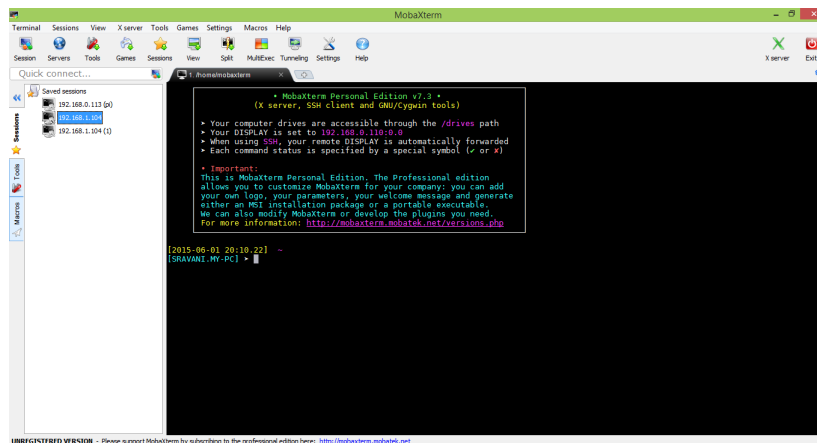
Note: A linux user needn't download any Xterm file. They can directly start accessing the R-Pi using the command: ssh -X pi@192.168.0.4 (IP address of pi) and then entering the login id and password for using the Pi remotely.

7 References

1. `http://en.wikipedia.org/wiki/Secure_Shell`

2. `http://www.webopedia.com/TERM/S/SSH.html`

3. `http://www.codemastershawn.com/library/tutorial/images/ ssh.tunnel.overview.gif`

4. `http://mobaxterm.mobatek.net/`

5. `http://www.suntimebox.com/raspberry-pi-tutorial-course/ week-3/day-5/`