



Adafruit's Raspberry Pi Lesson 6. Using SSH

Created by Simon Monk

A screenshot of a terminal window titled "pi@raspberrypi: ~". The window has a blue title bar and standard window controls. The terminal text shows a successful SSH login for user 'pi' from IP '192.168.1.13'. It displays the system version 'Linux raspberrypi 3.2.27+ #250 PREEMPT Thu Oct 18 19:03:02 BST 2012 armv6l', the Debian GNU/Linux welcome message, and the last login time 'Mon Dec 17 10:59:46 2012 from 192.168.1.6'. The prompt 'pi@raspberrypi ~ \$' is shown with a green cursor.

```
pi@raspberrypi: ~
login as: pi
pi@192.168.1.13's password:
Linux raspberrypi 3.2.27+ #250 PREEMPT Thu Oct 18 19:03:02 BST 2012 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Dec 17 10:59:46 2012 from 192.168.1.6
pi@raspberrypi ~ $
```

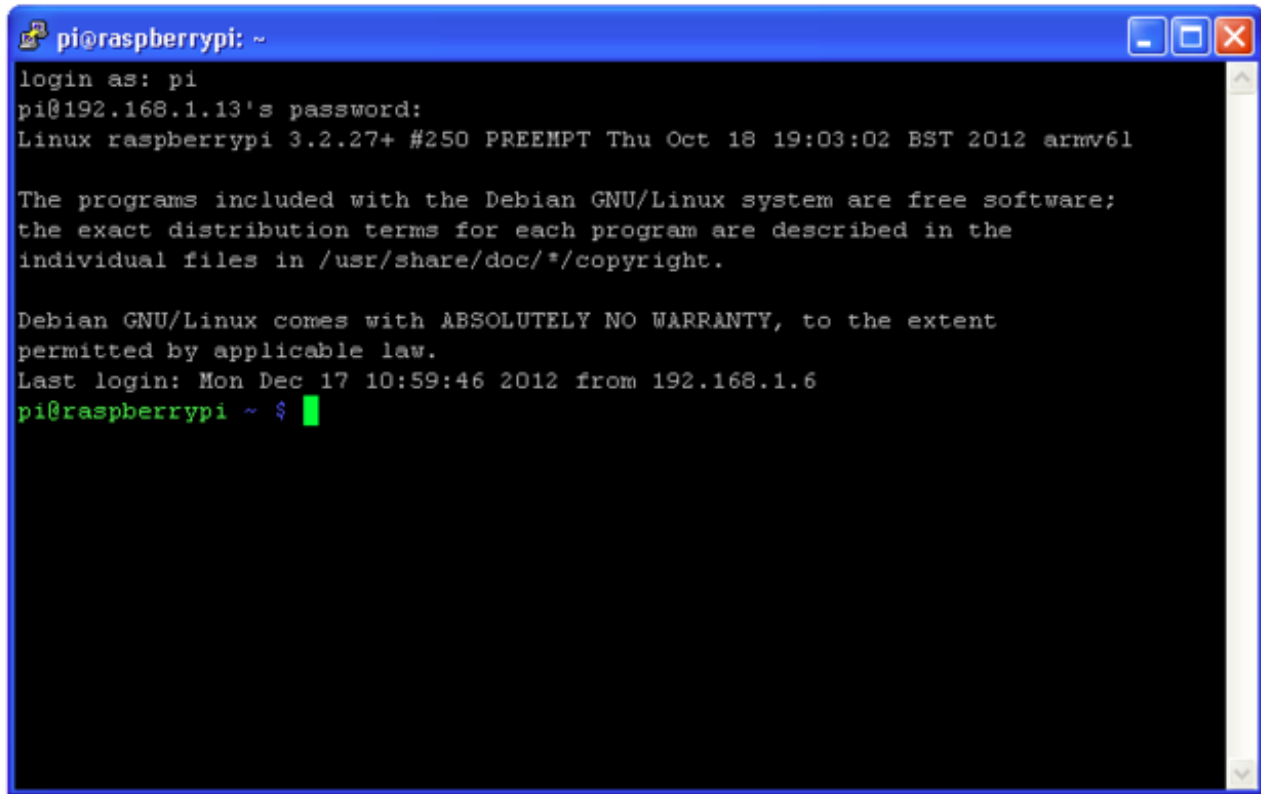
Last updated on 2017-08-16 01:12:07 AM UTC

Guide Contents

Guide Contents	2
Overview	3
Enabling SSH	4
Using a blank boot file	4
Using Raspi-Config	4
Using SSH on a Mac or Linux	8
SSH under Windows	10
Test & Configure	12
Troubleshooting	12

Overview

In this lesson you will learn how to remote control your Raspberry Pi over your local network using Secure Shell (SSH).



```
pi@raspberrypi: ~
login as: pi
pi@192.168.1.13's password:
Linux raspberrypi 3.2.27+ #250 PREEMPT Thu Oct 18 19:03:02 BST 2012 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Dec 17 10:59:46 2012 from 192.168.1.6
pi@raspberrypi ~ $
```

A common reason for remote controlling your Pi from another computer (like your laptop or desktop) is that you may be using your Pi solely to control some electronics and therefore not need a keyboard, mouse and monitor, other than for setting it up.

It also can just save on desktop clutter, and the problem of having multiple keyboards and mice all over the place.

Enabling SSH

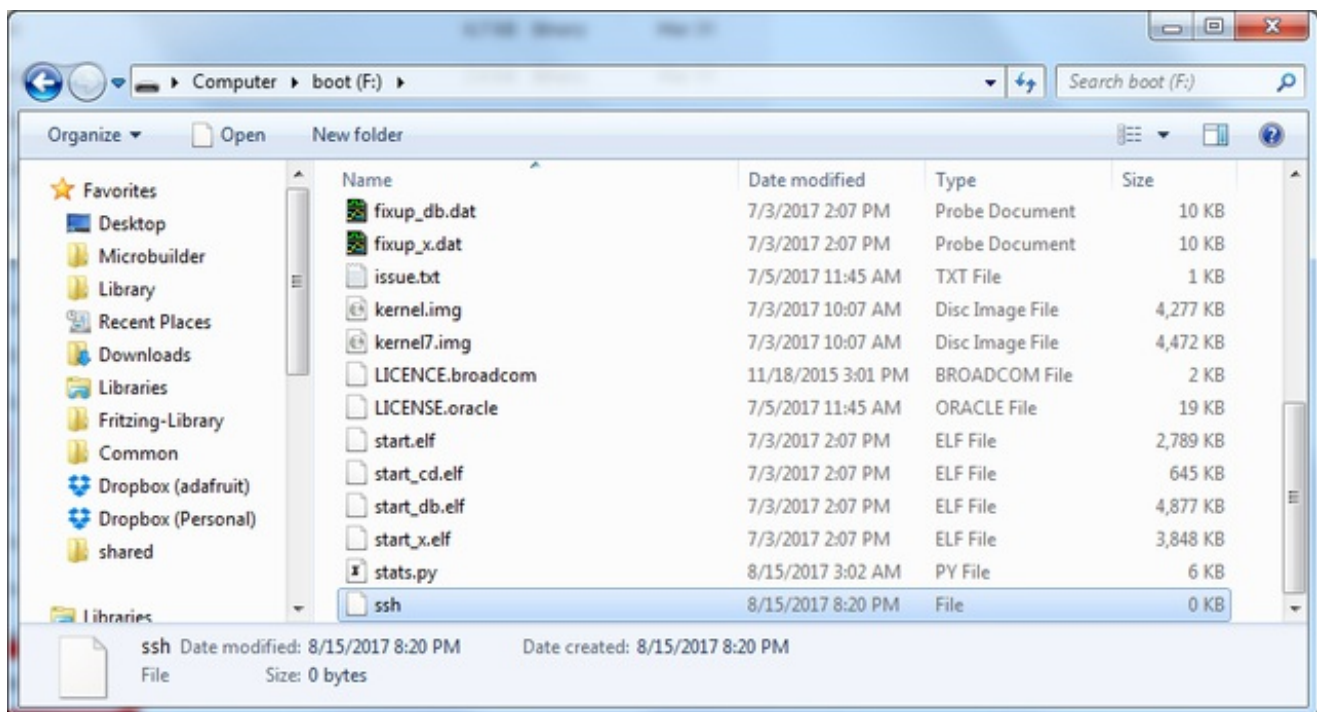
Secure Shell (SSH) is a feature of Linux that allows you to effectively open a terminal session on your Raspberry Pi from the command line of your host computer.

Recent versions of Raspbian do not enable SSH access by default. You can use an empty boot file or **raspi-config**, which you first saw [back in Lesson 2](http://adafru.it/aUa) (<http://adafru.it/aUa>)

Using a blank boot file

For truly headless setups, if you can't **ssh** into your Pi you can't turn on **ssh**!

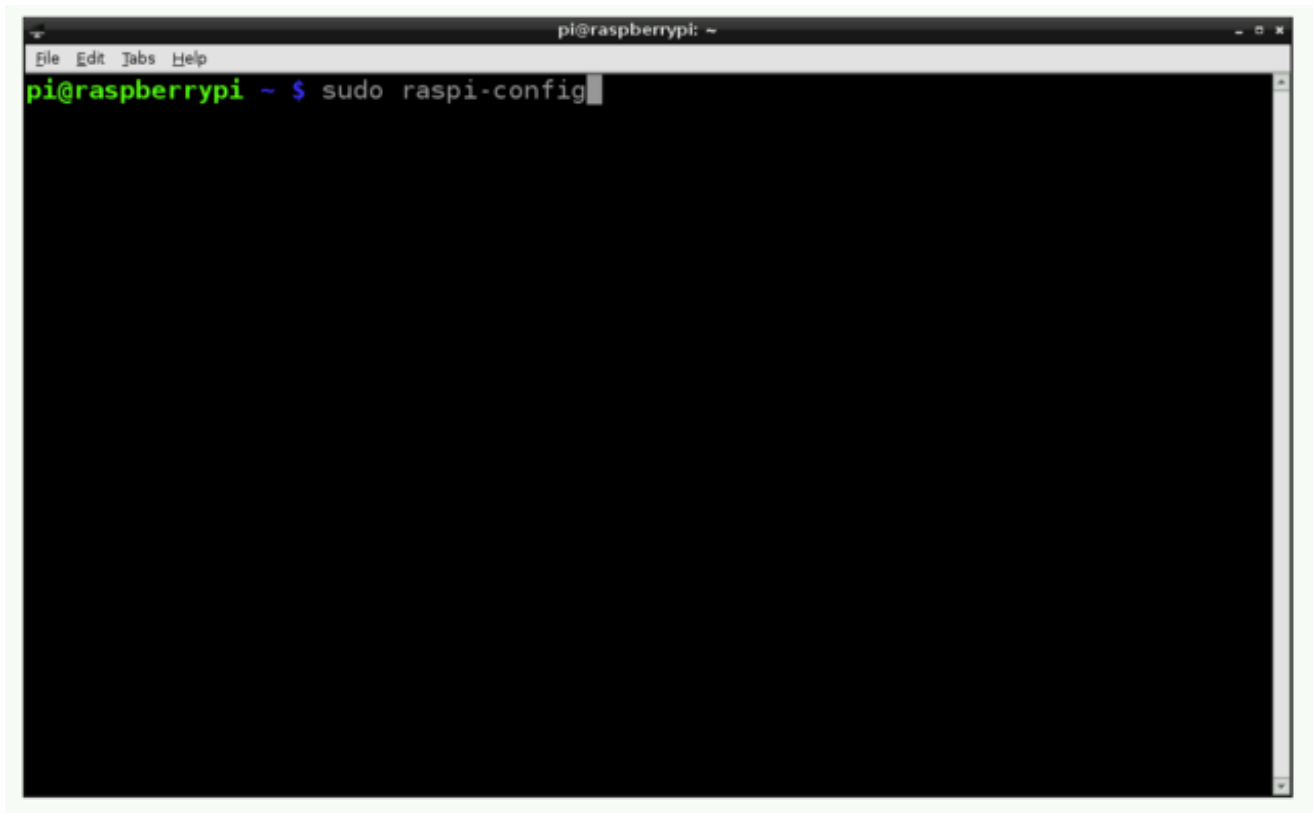
It's a bit of conundrum! But you can easily get around it by using a trick in Raspbian. To do so, we simply create a file called **ssh**. *This file does not exist by default and needs to be created.* It can be empty. The system looks for it at boot time and will enable ssh if it is there. It is then deleted. So just create a new file and save it as **ssh** to the **boot** folder. If you plug the SD card into your computer, just put that **ssh** file directly in the SD card director's root directory



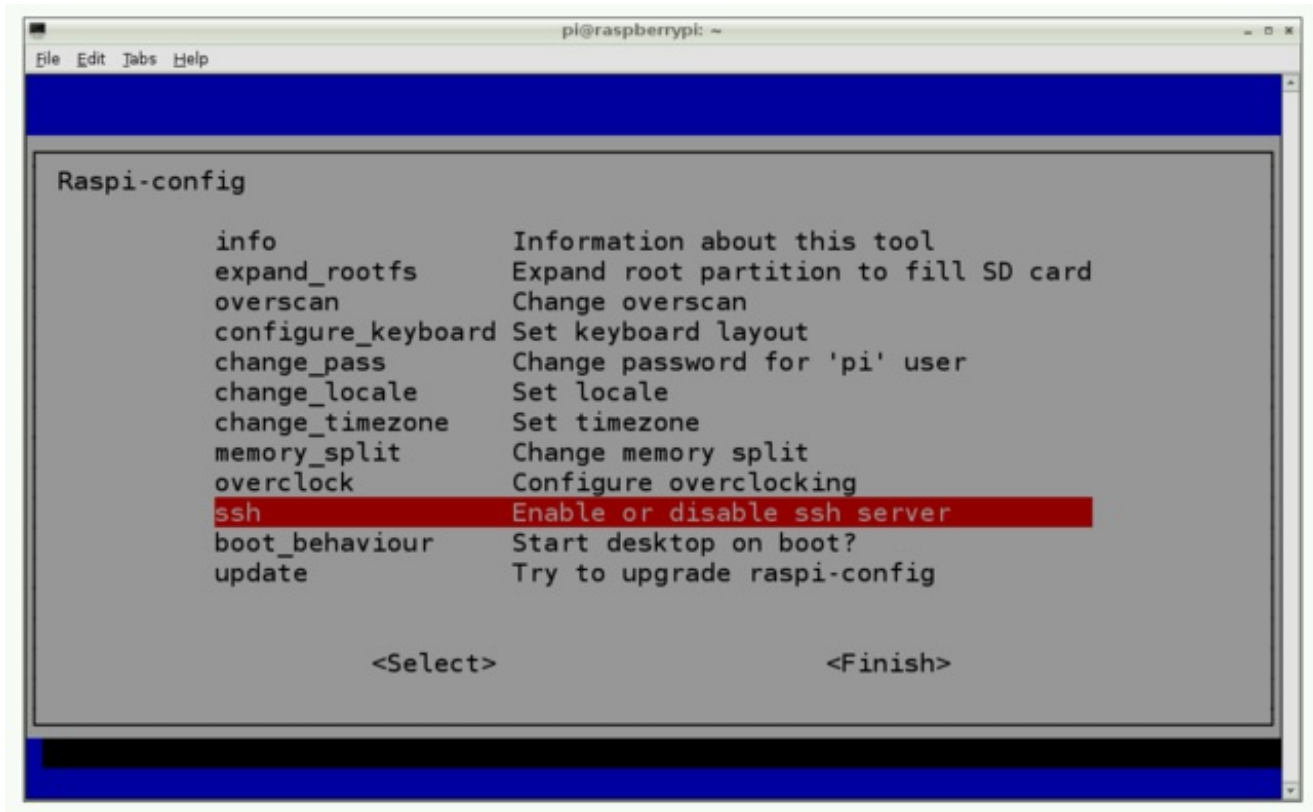
Using Raspi-Config

In order to do this, open LX Terminal on your Pi and enter the following command to start Raspi Config:

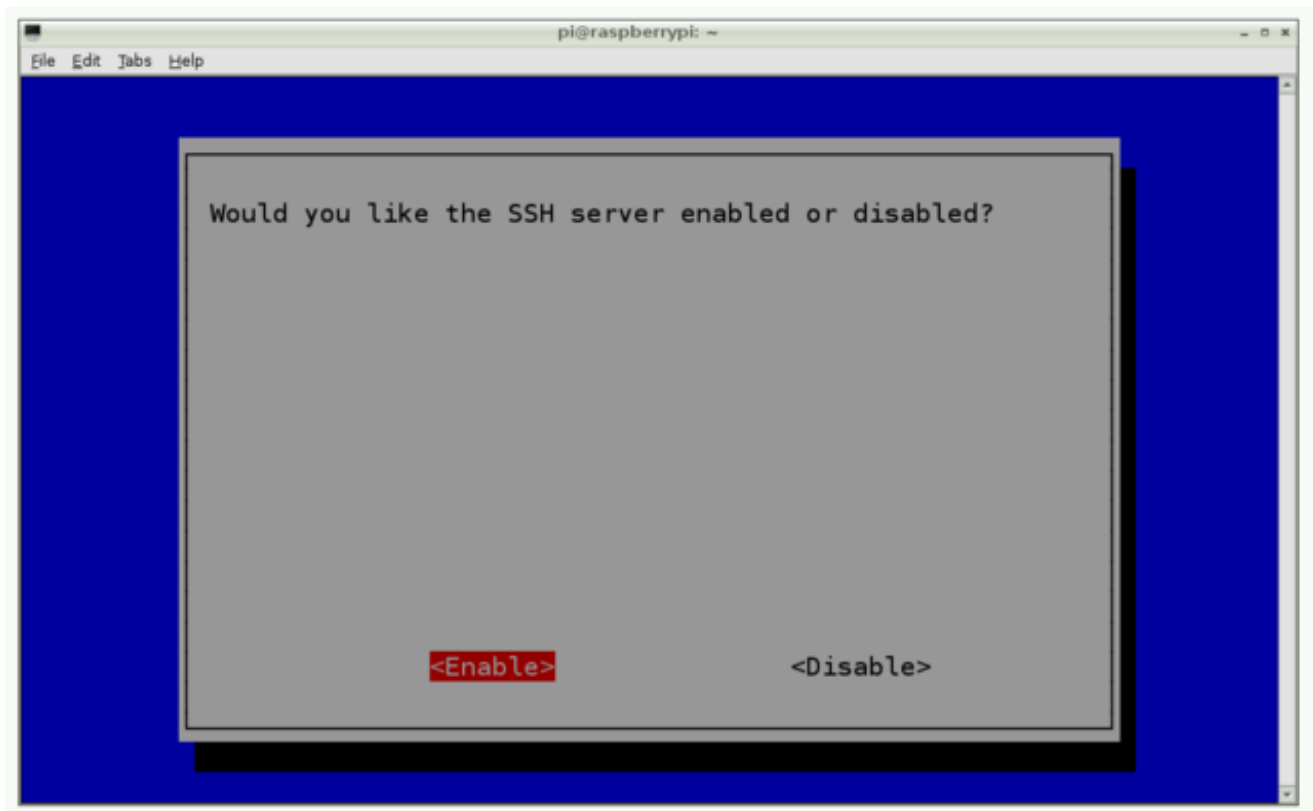
```
sudo raspi-config
```



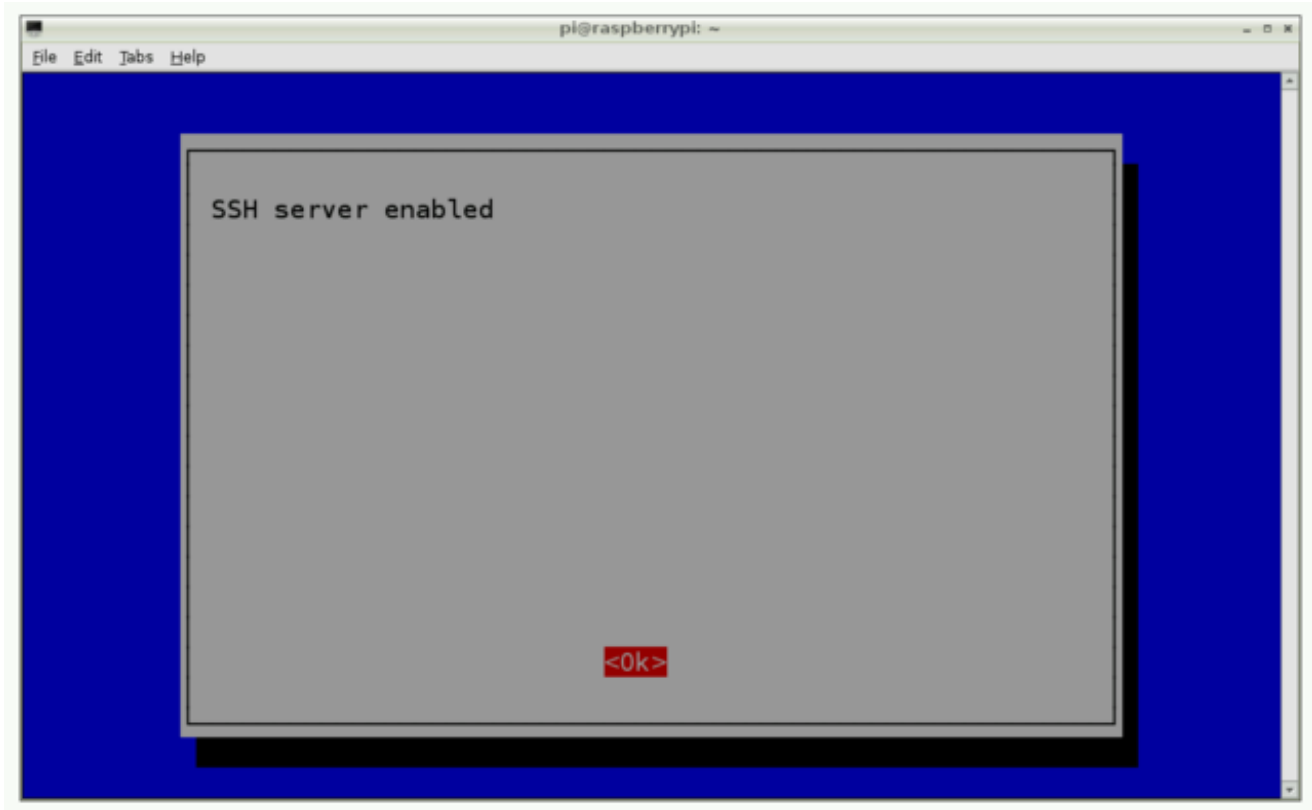
Scroll down to the “ssh” option, it might be under **Interfaces** or **Advanced** (they move it around)



Hit the Enter key and then select "Enable"



A script will run and then you will see the following as confirmation:



You will need to reboot your Pi to make the change permanent

Using SSH on a Mac or Linux

Now switch over to using the computer from which you wish to control the Pi.

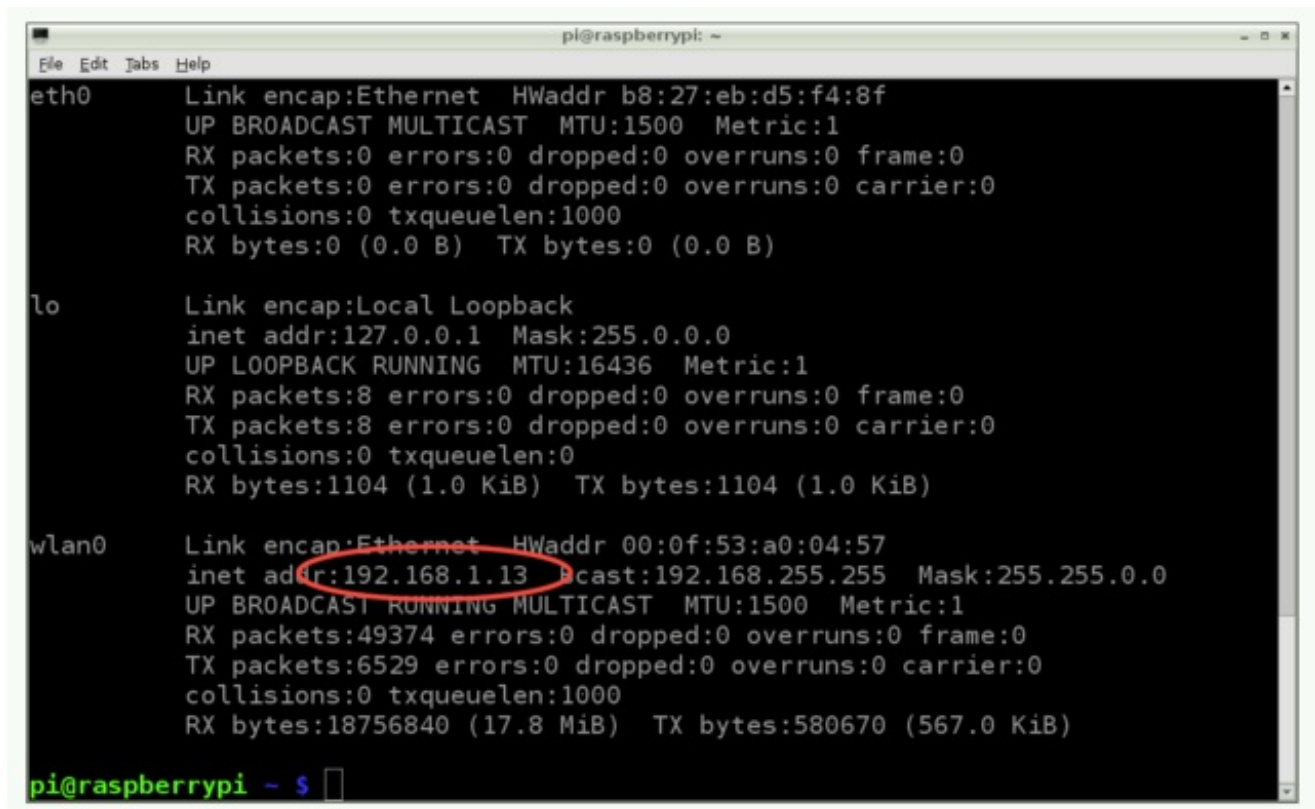
If you are using a Mac or Linux PC then open a Terminal. On the Mac, you can find this in the Utilities folder of your Applications folder.

Enter the following command into the Terminal window. *That's a lowercase L after the dash!*

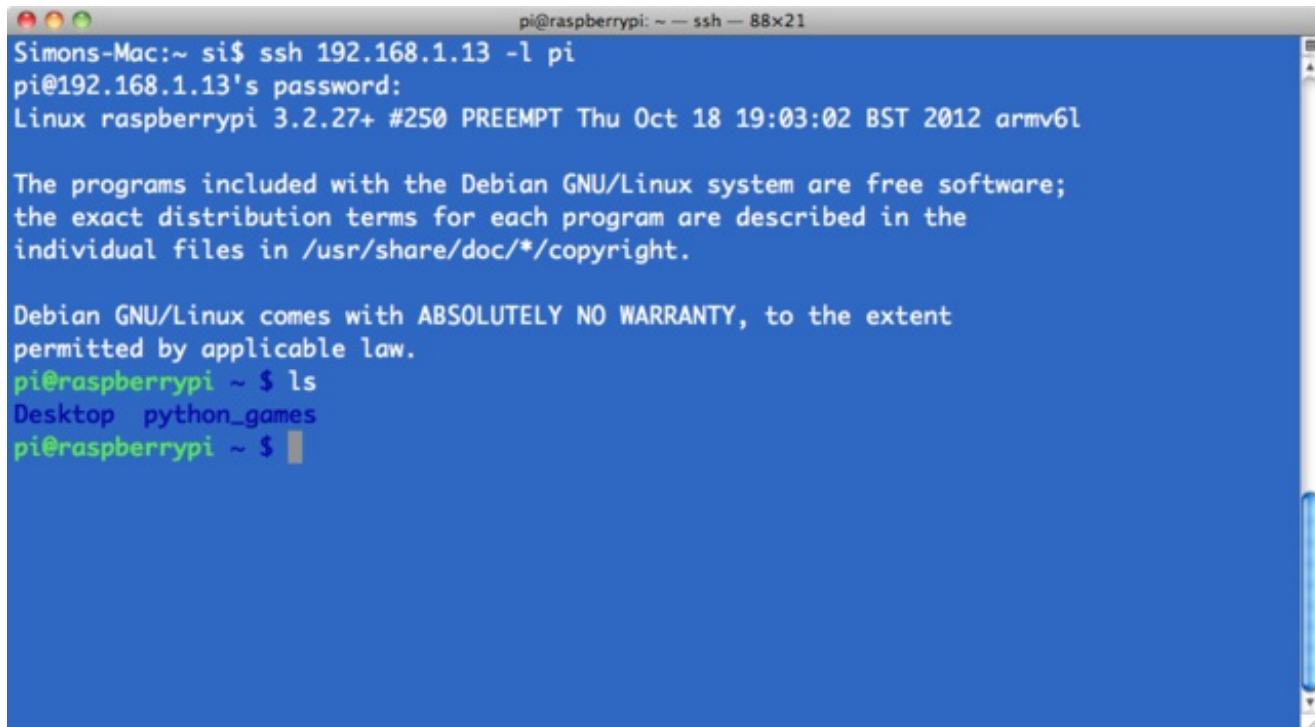
```
ssh 192.168.1.13 -l pi
```

You can also use `ssh pi@192.1691.13`

Note that you will need to replace the IP address above with that of your Pi. You can find this by running the command “`sudo ifconfig`” from the Terminal.



```
pi@raspberrypi: ~  
File Edit Tabs Help  
eth0      Link encap:Ethernet  HWaddr b8:27:eb:d5:f4:8f  
UP BROADCAST MULTICAST  MTU:1500  Metric:1  
RX packets:0 errors:0 dropped:0 overruns:0 frame:0  
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:1000  
RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)  
  
lo        Link encap:Local Loopback  
inet addr:127.0.0.1  Mask:255.0.0.0  
UP LOOPBACK RUNNING  MTU:16436  Metric:1  
RX packets:8 errors:0 dropped:0 overruns:0 frame:0  
TX packets:8 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:0  
RX bytes:1104 (1.0 KiB)  TX bytes:1104 (1.0 KiB)  
  
wlan0     Link encap:Ethernet  HWaddr 00:0f:53:a0:04:57  
inet addr:192.168.1.13  Bcast:192.168.255.255  Mask:255.255.0.0  
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1  
RX packets:49374 errors:0 dropped:0 overruns:0 frame:0  
TX packets:6529 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:1000  
RX bytes:18756840 (17.8 MiB)  TX bytes:580670 (567.0 KiB)  
  
pi@raspberrypi ~ $
```

```
pi@raspberrypi: ~ — ssh — 88x21
Simons-Mac:~ si$ ssh 192.168.1.13 -l pi
pi@192.168.1.13's password:
Linux raspberrypi 3.2.27+ #250 PREEMPT Thu Oct 18 19:03:02 BST 2012 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
pi@raspberrypi ~ $ ls
Desktop  python_games
pi@raspberrypi ~ $
```

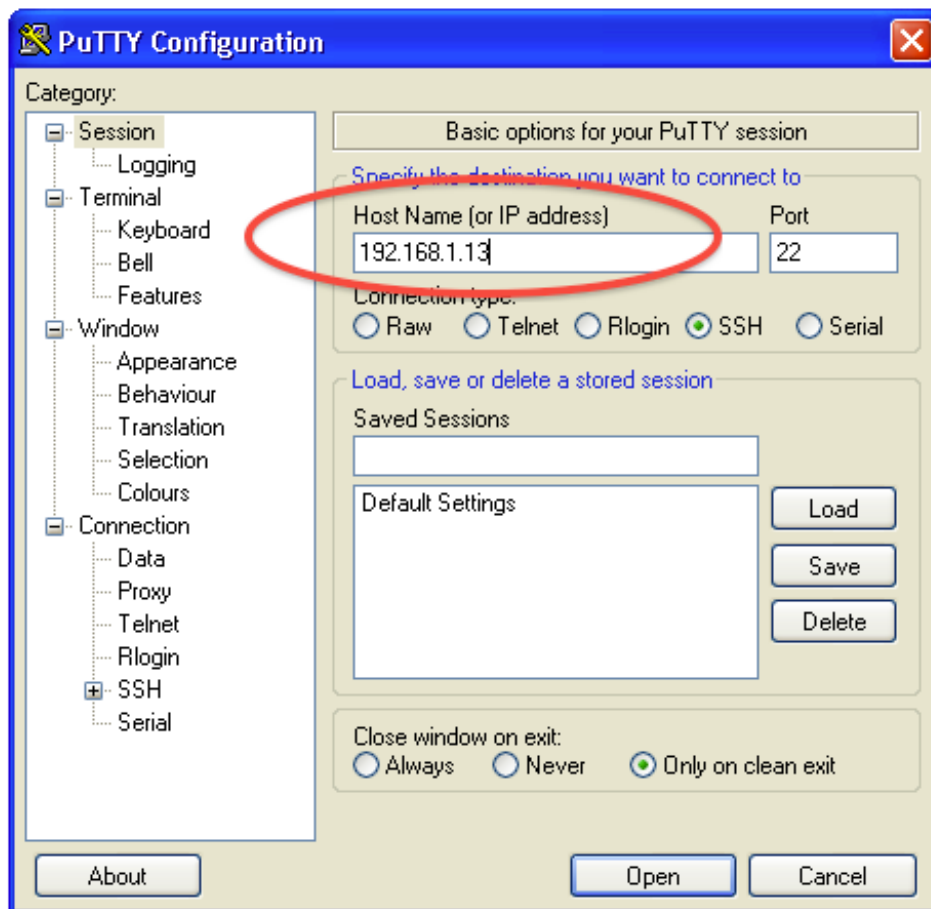
The option “-l pi” specifies that we want to log into the Pi as the user “pi”. The first time you run the command, you will get a security warning about not being able to verify the identity of the machine, say that you want to continue and enter your password (“raspberrypi” by default) when prompted.

You will notice that the command prompt will change to indicate that you are now connected to your Pi. Try using the “ls” command to show the contents of the current folder on the Pi.

SSH under Windows

If you use windows, then you will need to download a free program called “PuTTY” from here: <http://www.putty.org/> (<http://adafru.it/aUb>).

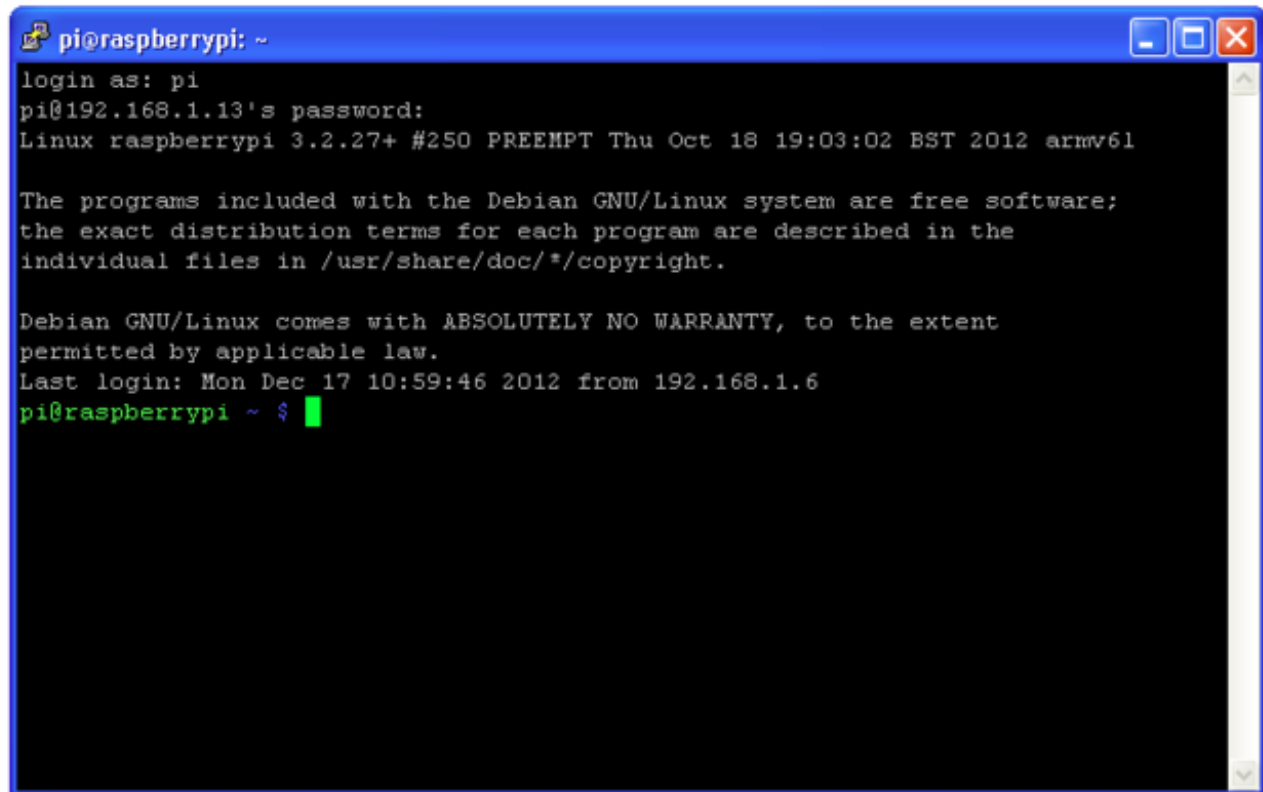
Having downloaded and installed PuTTY (it's a single file called putty.exe), run the program.



Enter the IP address that you found earlier and click “Open”. This will give you a warning (the first time) and then prompt you for the user (“pi”) and password (“raspberr”).



The ssh window will then be ready for use.

A terminal window titled "pi@raspberrypi: ~" showing the SSH login process. The text displayed is: "login as: pi", "pi@192.168.1.13's password:", "Linux raspberrypi 3.2.27+ #250 PREEMPT Thu Oct 18 19:03:02 BST 2012 armv6l", "The programs included with the Debian GNU/Linux system are free software; the exact distribution terms for each program are described in the individual files in /usr/share/doc/*/copyright.", "Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.", "Last login: Mon Dec 17 10:59:46 2012 from 192.168.1.6", and the prompt "pi@raspberrypi ~ \$" with a green cursor.

For a good example of how to use ssh to remotely configure a raspberry Pi, take a look at this tutorial:

<http://learn.adafruit.com/raspberry-pi-e-mail-notifier-using-leds/overview> (<http://adafru.it/aUc>)

Test & Configure

Try exploring your files system by using 'ls' to list the files in the current directory and 'cd' followed by a directory name to change the current directory.

You can edit files using 'nano' followed by the file name and also install software using the 'apt-get' command, as described in some of the earlier tutorials in this series.

When finished with your ssh session, close the client application/window or simply type in **exit** into the shell window.

Troubleshooting

If you encounter a **connection reset by peer** error when trying to connect to your Pi, there could be a problem with the SSH keys. You can 'reset' the keys with the following commands.

First, remove the old keys:

```
sudo rm /etc/ssh/ssh_host_*
```

Then regenerate them

```
sudo dpkg-reconfigure openssh-server
```

Then try again!