

Setting up your Raspberry Pi 3 Model B

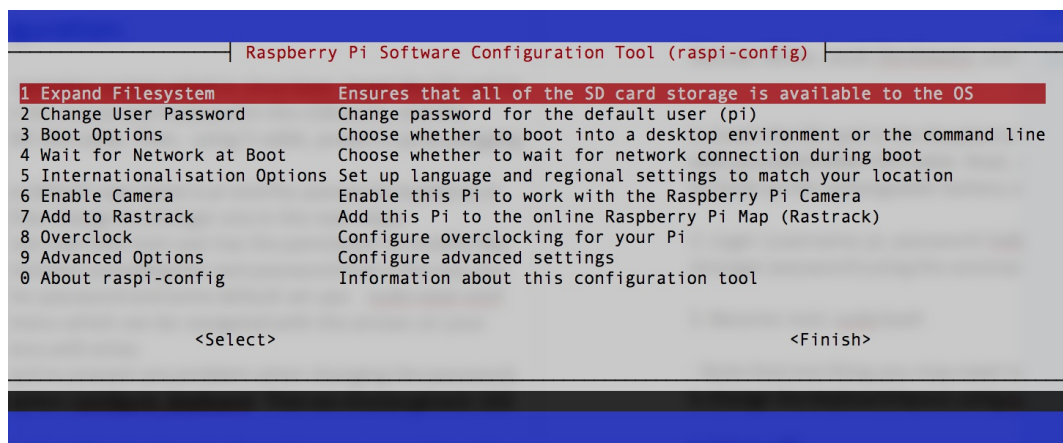
Starting Set up

- Each team should get a monitor, keyboard and mouse.
- Insert the mini SD card in the Raspberry Pi , connect the mouse and keyboard in the USB ports, and the display with provided HDMI-DVI cable. Then, using Y-cable, power it up by plugging in the charger.
- The operative system starts. Then, click on the **terminal** icon next to the menu to open it.
- Then you need to run some commands on it as root user to configure the Raspberry Pi (RPI). The **root user** has the permission to modify files or default settings as administrator providing the root password. The **root user** is **pi** and the default **root password** is **raspberrypi**. First we will make stronger the password, but first we will change some default set ups:

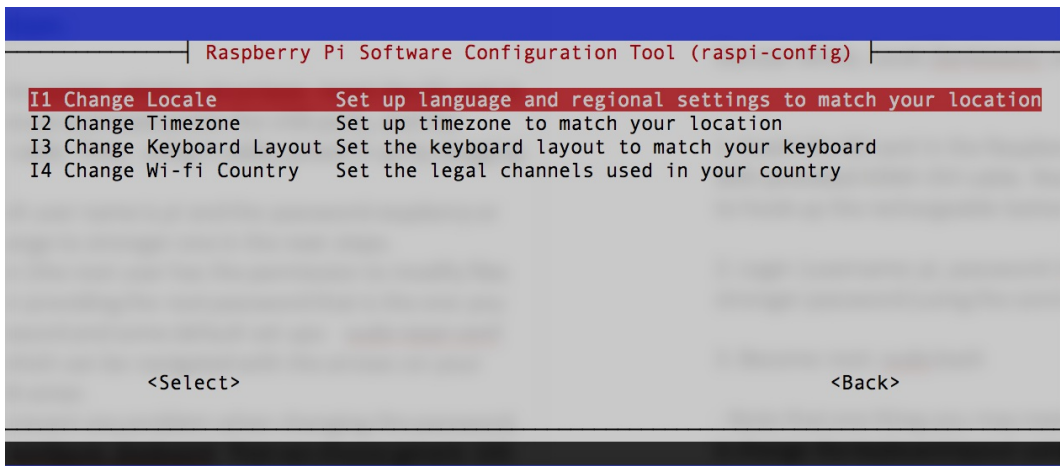
```
$ sudo raspi-config
```

Note: To execute any Linux command as root user, the *sudo* command presides the Linux command.

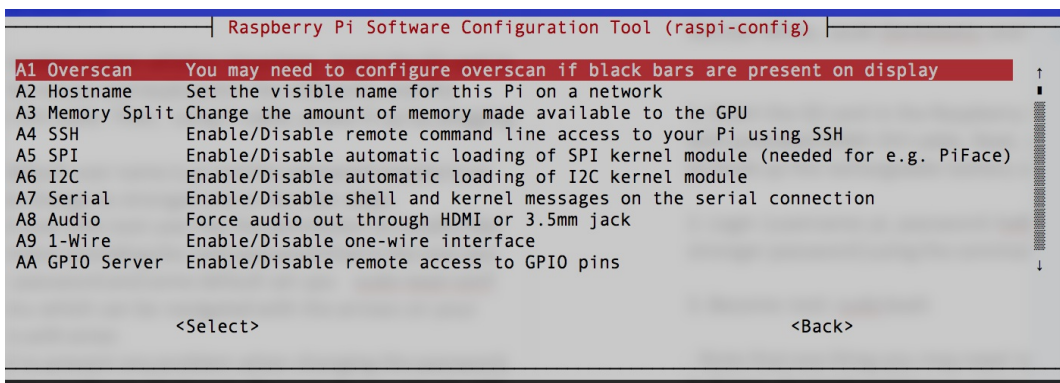
- The terminal will show a menu which can be navigated with the arrows on your keyboard and accept options with enter.



- Then we set up the keyboard to prevent any problem when changing the password, therefore we access the option: **Internationalisation Options** → **Change the Keyboard Layout**. Then we choose generic 105 key, and then UK.
- Also we have option to change the timezone from this menu.



- To change the password, we return to the main menu and choose the second option. We have to set the new password and do not reboot the RPi yet.
- We check that the [ssh](#) for remote network communications is enabled (security shell cryptographic network protocol). We access to the **Advance Options** → **SSH**.



- An optional step is to change the hostname in the same advance menu.
- Then we restart the RPi.

Setting WiFi from Imperial College network

- You will see there is no IP assigned to our PRI, therefore to set up the WiFi we need to modify a configuration file, but first we need to back it up:

```
$ sudo cp /etc/wpa_supplicant/wpa_supplicant.conf
/etc/wpa_supplicant/wpa_supplicant.conf_backup
```

Then we will edit the *wpa_supplicant.conf*:

```
$ sudo nano /etc/wpa_supplicant/wpa_supplicant.conf
```

The default text editor installed in the RPi is *nano*. We can install also *vim* or *vi* to have another option of a text editor.

This file should just have the next line at the beginning:

```
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
```

Then we add to the content of *wpa_supplicant.conf* the lines after # IC (**the configuration is case sensitive, so make sure you do not have typos**):

```
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1

# IC
network={
    ssid="Imperial-WPA"
    proto=RSN
    key_mgmt=WPA-EAP
    pairwise=CCMP
    auth_alg=OPEN
    eap=PEAP
    identity="ic\COLLEGE_USERNAME"
    password="YOUR_PASSWORD"
}
```

This is the Imperial College configuration in which you have to replace "COLLEGE_USERNAME" with a valid college username, please do not store your password in plain text, but we will change it after verifying that the WiFi is working. Reboot the system if it is necessary. **Note:** In case you are working with your team, and you do not want to show your password, just leave the field blank and follow the next steps to encrypt your password before setting it up in the *wpa_supplicant.conf*.

Encrypting Password

In order to not store the password in a plain text we **encrypt** our password with an **MD4 hash generated** from the corresponding college password. You can generate the hash like this with the next Linux command:

```
$ echo -n 'YOUR_PASSWORD' | iconv -t utf16le | openssl md4
```

This command will display the encrypted password on your terminal like:

```
$ (stdin)= a6c71eedc2eacbca84003336a4a62a1c
```

Then you can copy the string that was generated in your terminal screen similar to the one of the example above: a6c71eedc2eacbca84003336a4a62a1c.

Note: Also you can save the hash from your password in a file and then read its content:

```
$ echo -n 'YOUR_PASSWORD' | iconv -t utf16le | openssl md4 > hash.txt
$ cat hash.txt
```

The *cat* command is used to read and concatenate files.

Then open the *wpa_supplicant.conf* to add the hashed password you generate:

```
$ sudo nano /etc/wpa_supplicant/wpa_supplicant.conf
```

In the password field replace "YOUR_PASSWORD" with the string you generated as hexadecimal characters, and we add the 'hash: '-prefix) in the similar fashion as in the example below:

```
# IC
network={
    ssid="Imperial-WPA"
    proto=RSN
    key_mgmt=WPA-EAP
    pairwise=CCMP
    auth_alg=OPEN
    eap=PEAP
    identity="ic\COLLEGE_USERNAME"
    password=hash:a6c71eedc2eacbca84003336a4a62a1c
}
```

A last security step to do is to remove the bash history, with all the commands we had typed on the terminal. Therefore, we do it like:

```
$ history -w
$ history -c
```

Then we reboot again the RPi to check that the password was properly set up.

Installing Python

To install Linux packages in our system we must use the command: `sudo apt-get install name_of_package`. The installation could take some minutes.

Updating operative system

```
sudo apt-get update
```

Installing C lib needed by Python:

```
sudo apt-get -y install libffi-dev
sudo apt-get -y install libssl-dev
```

Installing Python:

```
sudo apt-get -y install build-essential python-dev python-openssl
sudo apt-get -y install python-setuptools
sudo apt-get -y remove --purge python-pip
sudo apt-get -y install python-pip
sudo pip install --upgrade pip
```

Installing other text editor:

```
sudo apt-get -y install vim
```

Remote connection to your Raspberry Pi

Weaved services connect you easily and securely to your Pi from a mobile app, browser window and a terminal. Control remote computers using tcp hosts such as SSH. You will be able to connect to your RPi from laptop or desktop at home. The free weaved account allows for 10 registered services and 30 minute connections on up to 1

concurrent service(s).

Installing weaved:

Manage network devices remotely using [weaved](#) service. To install:

```
sudo apt-get -y install weavedconnectd
```

Weaved configuration:

- To configure weaved in our RPi, first we need to open an account in [weaved](#) website. You can register from your laptop or desktop. Once you have it, from your Rpi terminal you will execute a command to link your RPi to your weaved account:

```
sudo weavedinstaller
```

- Enter your Weaved account username and password. Next, you will see this menu:

```
=====
Protocol      Port      Service      Weaved Name
=====

***** Main Menu *****
*
*      1) Attach/reinstall Weaved to a Service      *
*      2) Remove Weaved attachment from a Service  *
*      3) Exit                                       *
*
*****
Please select from the above options (1-3):
█
```

- Initially you won't have any Weaved services installed, so the upper part is empty. Enter **1** to attach Weaved to an existing TCP service (host) on your Raspberry Pi. You should now see the following screen:

```
=====
Protocol      Port      Service      Weaved Name
=====

***** Protocol Selection Menu *****
*
*      1) SSH on default port 22                    *
*      2) Web (HTTP) on default port 80             *
*      3) VNC on default port 5901                 *
*      4) Custom (TCP)                             *
*      5) Return to previous menu                  *
*
*****
Please select from the above options (1-5):
█
```

Enter **1** for SSH.

- Next, we accept the default port (**y**).

```
The default port for SSH is 22.
Would you like to continue with the default port assignment? [y/n]
```

- The installer confirms your choice and asks you to give this connection a name (you can make it up, but remember to make a name easy for you to identify a specific RPi in case you have more than one attached to the weaved service.):

```
We will attach a Weaved connection to the following service:

Protocol: ssh Port #: 22

Enter a name for this Service Attachment (e.g. SSH-Pi).
This name will be shown in your Weaved Device List.

Raspberry-Pi-SSH
```

You will now return to the main menu, where you can see your Weaved Service Connection installed, then enter **3** to exit.

```
=====
Protocol      Port      Service      Weaved Name
=====
SSH           22        sshd         Raspberry-Pi-SSH

***** Main Menu *****
*
*      1) Attach/reinstall Weaved to a Service      *
*      2) Remove Weaved attachment from a Service  *
*      3) Exit                                       *
*
*****
Please select from the above options (1-3):

```

Accessing from your laptop

We will see here how you can access using your laptop or any other desktop from any terminal. First, if you login to your weaved account, you will get a list of the services linked to your devices:

Your current list of services

Click on service names to connect. Your account allows for 10 registered services and 30 minute connections on up to 1 concurrent service(s). You can [Upgrade Now](#) to get more services, longer connection times and more concurrent connections.

Name	Type	Status	
raspberrypiSSH	SSH	online	Share Settings
raspberrypiWeb	HTTP	online	Share Settings

In your case you will have just one item with a Type SSH as in the first line at the screen shoot above. When you click on the name of your device, your browser will open and show you a window like:

Remote SSH Connection

your service

Use a browser Plug-in for single click connection
[FireSSH](#) [Get the plug-in](#)

Or, copy and paste the values below to your SSH application:

Or, copy and paste one of these command lines into your terminal window, based on your SSH username:

For pi username	<input type="text" value="ssh -l pi proxy71.weaved.com -p 34644"/>
For root username	<input type="text" value="ssh -l root proxy71.weaved.com -p 34644"/>
All others *	<input type="text" value="ssh -l LOGIN proxy71.weaved.com -p 34644"/>

* Replace LOGIN with your device login name.

[Click here](#) for additional help

Then we copy the command after *For pi username*, in this example it is: `ssh -l pi proxy71.weaved.com -p 34644`. For you it will be different. Then, paste the command in your laptop or desktop terminal (If you are using a Mac or Linux all will work, but for windows you have to [install a SSH and Telnet client](#).),

Then, you are connected from your laptop to your RPi!! You don't need the display and mouse anymore!

Virtual terminal

Remember you can just be connected during 30 minutes using **weaved**, after that time you have to connect again to your account and do the same procedure we explained in the previous section. Therefore we will show you how a *virtual terminal* can help you when you are working on your RPi.

Screen is a full-screen software program allows you to use multiple windows (virtual VT100 terminals) in Unix. It offers a user to open several separate terminal instances inside a one single terminal window manager.

The screen application is very useful, if you are dealing with multiple programs from a command line interface and for separating programs from the terminal shell. It also allows you to share your sessions with others users and detach/attach terminal sessions.

When to use screen?

One of the advantages of *Screen*, is that you can detach it. Then, you can restore it without losing anything you have done on the *Screen*. One of the typical scenario where *Screen* is of great help is when you are in the middle of SSH session and you want to download a file, update the operative, or transfer a big file to your RPi. The process could be 2 hours long. If you disconnect the SSH session, or suddenly the connection lost by accident, then the download process will stop. You have to start from the beginning again. To avoid that, we can use screen and detach it.

Installing screen

The screen program allows you to use multiple windows (virtual VT100 terminals) in Unix. If your local computer crashes, or you are connected remotely and lose the connection, the processes or login sessions you establish through screen don't get lost.

```
sudo apt-get -y install screen
```

Some Examples to use screen

- When you are in your terminal, you can create a *screen* or virtual terminal e.g. we will name the screen *mysession*:

```
pi@RPi-Dendrite:~ $ screen -S mysession
```

- Then you will be automatically attached to your screen or virtual terminal, that from now on we will call just *screen*. You can execute now commands and work in the terminal:

```
root@RPi-Dendrite:/home/pi# echo "hello world"
hello world
root@RPi-Dendrite:/home/pi#
```

- You can detach from the *screen* by pressing "Ctrl-A" and "d". Then we will go to our terminal outside the *screen* session. Then we can check the list of *screens*:

```
pi@RPi-Dendrite:~ $ screen -ls
There is a screen on:
  29097.mysession (14/10/16 12:23:55)      (Detached)
1 Socket in /var/run/screen/S-pi.
pi@RPi-Dendrite:~ $
```

- We get the list and the screen ID. If we want to attach to a particular *screen*:

```
pi@RPi-Dendrite:~ $ screen -r 29097
```

Basic commands to work with the virtual terminals

Screen command	Description
<code>screen -S name_of_terminal</code>	Assigning name to the virtual terminal or screen session.
<code>screen -ls</code>	List all the virtual sessions or screens opened.
<code>screen -X -S name_of_terminal quit</code>	Kill an specific virtual terminal.
<code>screen -r name_of_terminal</code>	Attach to the virtual terminal or screen.

Press "Ctrl-A" and "d"	Detach from virtual terminal or screen.
Press "Ctrl-A" and "K"	This command will leave and kill the virtual terminal or screen
Press "Ctrl-A" and "n"	Switching to the next virtual terminal or screen.
Press "Ctrl-A" and "p"	Switching to the previous virtual terminal or screen

For more examples go to the [link](#) or ask the instructors.

To know more about more advance details of how connect remotely go to the [advance guide](#):