How to setup a Raspberry Pi 3/3b/4/4b

Dustin Powell

IT415 - 01

System Administration 1

Fall 2019

Abstract

The purpose of this manual is to provide a step by step process to properly setup a Raspberry Pi with the Linux Kernel Raspbian version 4.19. The entire setup process occurred on two Raspberry Pis one being a Raspberry Pi 4b and a Raspberry Pi 3b. The entire process from installing Raspbain to setting up cron jobs to for backing up user files. Each of these task were completed on the following listed days.

10/24/19 - A brief outline of steps were created and inputted into the beginnings of a Latex document

10/29/19 - The steps from setting up the Raspberry Pi to checking the the operating system were completed.

10/31/19 - The steps to add new users and setup their permissions were finished. The Raspberry Pi was also connected to the internet to install updates.

11/5/19 - The hosts.allow and hosts.deny files are created to secure the Raspberry Pi at the application level from intrusion. SSH server setup is also complete to allow for remote access.

11/7/19 - Title, sections, subsections, and proof reading of current content is completed. 11/

11/16/19 - RDP through the Apache Web Servers steps of the manual are completed. An abstract is added to the document for a summary of the project.

11/19/19 - The Raspberry Pi is cloned and setup in the Lab for use. The last section is completed. Last proof read of manual is completed.

The following hardware resources were used for the example build:

1. Raspberry Pi 4b Cana Kit

2. Raspberry Pi 3b Cana Kit

3. SanDisk Ultra Plus 32 GB SD Card

4. SD Card that comes with Cana Kit with the Raspberry Pi 3b

Contents

1   Initial Setup

The first section covers collecting peripherals, installing the Raspbian operating system, verification of a successful installation, removing unwanted software, changing the hostname, setting up a WiFi connection, and accomplishing the first update.

1.1   Assembling the Raspberry Pi

1. First obtain a USB keyboard and mouse.

2. Obtain a computer monitor with an HDMI input port to display the desktop of the Raspberry Pi.

3. Now format a 32 gigabyte or larger micro SD card on another computer.

4. Download the latest version of the Raspbian kernel.

5. Copy the image file contained in the ZIP file to the Micro-SD card.

6. Insert the Micro-SD card into the Raspberry Pi. Once completed, the assembly of the Raspberry Pi is complete.

1.2   Installing Raspbian

1. Before booting the Raspberry Pi, make sure that all peripherals, the monitor, power cable, and Micro-SD are properly connected to the Raspberry Pi.

2. Turn on the Raspberry Pi. As the kernel boots for the first time select to install the Rasbian kernel when prompted.

3. As the Raspberry Pi is creating partitions and installing Raspbian do not turn off the Raspberry Pi because it can corrupt the files contained on the Micro-SD card.

4. Follow the prompts for the installation of Raspbian.

5. Once the installation of Raspbian is completed and the login in screen loads. Login the Raspberry Pi with the login username `pi` and the password `raspberry`

6. Since the the pi user's default password is now very well known perform the following command:

```
passwd
```

Follow the prompts to create a new password for the Pi to help prevent unauthorized access to the device.

7. Once the Raspberry Pi, is fully booted login with the user `pi` and the new password that was just created.

1.3 Operating System and Hardware Verification

1. To begin verifying the installation of the operating system the kernel log need to be checked for any errors that have occurred. All systems logs can be found in the directory `/var/log`. To navigate to the directory perform the command:

```
cd /var/log
```

2. Once in the directory `/var/log` use the command:

```
ls -alt
```

The command `ls -alt` is very useful for locating and providing information about files and folders in a given directory.

3. Now check the kernel log file for any problems that have occurred with the operating system and hardware by performing the command:

```
sudo less kern.log
```

The `less` and `more` command are useful for viewing text files in the terminal. If no errors are found then continue to the next step.

4. The next step is to check the hardware temperatures of the Raspberry Pi to ensure that every component is working properly and the temperatures are in a safe operating range. The following command will display the temperature of the system.

```
sudo /opt/vc/bin/vcgencmd measure_temp
```

If the Raspberry Pi is in a safe temperature range then the system is performing normally.

1.4   Removing Programs and Adding Users

1. After checking the system is determined to running as normal any unwanted software should be found and removed from the system.

   ```
   sudo apt list
   sudo apt --purge remove PROGRAM NAME
   ```

   Next add the users to the Raspberry Pi using using the commands:

   ```
   sudo adduser USERNAME
   ```

2. Now follow the prompts to create a password for the new user.

3. When prompted to input user information press enter to skip the information.

4. Now that the user(s) have been created, now edit the user(s) to only allow access to the groups audio and video using the following command:

   ```
   sudo usermod -aG audio USERNAME
   sudo usermod -aG video USERNAME
   ```

1.5   Setting up a Wifi Connection

1. Now user the rasp-config to setup a wireless network connection

   ```
   sudo raspi-config
   ```

2. Select network options and then select hostname then follow the prompts to pick a desired name of the Raspberry Pi to display on a network.

3. Now select network options again and then select Wifi to setup the wireless connection for the device. Follow all prompts to setup the connection.

1.6 Performing Updates

1. Now perform the following commands to update the kernel and various software:

   ```
   sudo apt-get update
   sudo apt-get upgrade
   ```

   Follow all prompts to complete the update process. Once this process is complete a system restart is recommend to ensure that all of the changes take place.

2 Networking and Remote Access Installation and Configuration

The next few steps will cover setting up the network security measures to help prevent unauthorized access to the raspberry pi from the remote access services that are to be installed. In the following steps before each configuration file is changed a copy of the base file is to be saved before any edits occur to the configuration file. The command is the following

```
cp SOURCE DESTINATION
```

2.1 Application Layer Network Security

1. The first thing to setup the next work will be to setup the hosts.allow and host.deny files for a specic IP address or range of IP address to access the Raspberry Pi. The files are located in the /etc directory of the file system. The /etc directory can be access by using the following command:

   ```
   cd /etc
   ```

2. Once in the directory the host.allow and host.deny files need to be edited which determine which IPs have access to programs located at the application layer of the networking stack of the raspberry Pi. The hosts.allow is read first by the system then the hosts.deny is read. Using the commands to access the files.

```
sudo nano hosts.allow
sudo nano hosts.deny
```

Do not forget to allow localhost and 127.0.0.1 in the hosts.allow file so that any program uses the connection can be used.

3. First edit the hosts.allow, then add the various services and IPs or IP ranges that will have access to the various remote access programs. The syntax as follows:

```
SERVICENAME : IP RANGE OR SPECIFIC IPs
```

(Note: For multiple IPs they are comma delimited and a range of IPs must be enter individually for each service)

4. Now edit the hosts.deny file, then add the various services and IPs or IP ranges that will not have access to the various remote access programs. The syntax is the same as the previous step.

5. Now perform the next command to go back to the starting directory.

```
cd
```

2.2 Secure Shell (SSH)

1. Now that security measures at the application level are set in place. Now we can install Secure Shell server, or SSH for short, to allow remote access to the raspberry pi via the command line. First use the command

```
sudo raspi-config
```

2. Navigate to interfacing and select the option. Then navigate to and select the SSH option to enable SSH for the Raspberry.

3. Next perform the following command to update the SSH Server and Client:

```
sudo apt-get install
```

This will install the lastest version of the SSH Server and update SSH Client to the latest version.

2.3   VNC Server

1. To install the VNC server uses the command to enable the VNC server just like activating SSH.

    ```
    sudo raspi-config
    ```

2. Select interfacing options, and then select VNC. Follow all prompts to setup the VNC sever for the Raspberry Pi.

3. Now type the command to generate a catchphrase and signature for the VNC server.

    ```
    vncserver
    ```

    Once this step is completed the realVNC server will launch on start-up on the Raspberry Pi, and since realVNC does not use TCP Wrappers the hosts.allow or hosts.deny do not need to be edited.

4. On another computer with a VNC viewer program, the Raspberry Pi can be controlled by the client computer. However, the IP address of the Raspberry Pi is needed to create a connection.

    ```
    ifconfig
    ```

    The IP address is listed under the `wlan0` ,next to `inet` is the current IP of the Raspberry Pi. This will be used by the VNC client to create a connection.

2.4   Remote Desktop (RDP)

Now install RDP, which is another program that can be used to access and use the the desktop on the Raspberry Pi. This program is very similar to VNC but different protocols are used for communication.

1. In the terminal do the following command to install RDP

    ```
    sudo apt-get install xrdp
    ```

2. To perform any edits to the xrdp server use the following command. By default the encryption level is set to high and and no TCP Wrappers are used by the program. So the file does not need to be edited.

```
sudo nano /etc/xrdp/xrdp.ini
```

On another computer with a RDP program, the Raspberry Pi can be controlled by the client computer. However, the IP address of the Raspberry Pi is needed to create a connection.

```
ifconfig
```

The IP address is listed under the `wlan0` ,next to `inet` is the current IP of the Raspberry Pi. This will be used by the RDP client to create a connection.

2.5 Apache Web Server

For this subsection install an Apache Web Server available to the World Wide Web is not recommend however, the web server can be very useful on a local network for various means. This section is a brief overview as many modules and configurations exist for an Apache Web Server on a local network.

1. To install the Apache Web perform the following command to install the server.

   ```
   sudo apt-get install apache2 -y
   ```

   Follow all prompts to install the server. Once completed, reboot is recommended to ensure that all changes are made.

2. After the reboot, login and open a web browser, and type the following IP address in the website address bar.

   ```
   127.0.0.1
   ```

   If the default web page comes up, the web server is working properly. If not then check to make sure the server is running and that the hosts.allow files has `ALL : localhost, 127.0.0.1`.

3. Now files located in `/var/www/html/` need ownership to be changed to allow edits to the web pages to occur. To do so perform the following commands:

   ```
   cd /var/www/html/
   ```

```
sudo chown USERNAME: FILE_NAME
```

4. To edit the Apache Web Server configuration files perform the following commands.

```
cd /etc/apache2
sudo nano FILENAME
```

The folder contains multiple configuration files and sub directories so use caution when changing the files to the desired specification.

5. Lastly, any additional packages used by the web server and be installed by performing the `apt-get` command at step one. Each package will vary between having configuration files and not so beware before installing any packages.

3 System backups via Rsync, Cron Jobs, and Cloning

This section discusses using RSync and Cron Jobs for setting up remote backups. The cloning section will cover mutiple ways to clone a Raspberry Pi.

3.1 RSync and Cron Jobs For File Back-ups

1. Rsync is used for backing up files to another location on a device or to a remote device using SSH. The first step is to install Rsync by using apt-get.

   ```
   sudo apt-get install Rsync
   ```

   Follow all prompts to complete the installation of Rsync.

2. Now we need to create a cronjob. The purpose of cron is to schedule system task based upon date and time. The first part of the crobjob to set is the time which follows the order : minute, hour, day of month, month, and day of the week. The next part follows the command that is to be carried out by cron. This can be either a bash script to run or just a single bash command. The first step is to open crontab to create the cronjob by using the following command.

   ```
   sudo crontab -e
   ```

3. Now enter the cron job into the file just like the following:

   ```
   0 0 * * 0 sudo rsync -av ssh /home username@IP_ADDR: /home/pibak
   ```

3.2 Cloning SD Cards

An SD card can be cloned various way either by command line or by a program with a GUI interface. Raspbian has a GUI interface program that provides a simple interface for duplicating a SD card. Which will be the method for cloning in this case.

1. Select the Raspberry Icon on the top left of the screen

2. Navigate to and select accessories

3. Now navigate to and select SD Card Copier

4. When prompted enter the super user password when prompted

5. Now select the source SD Card in the first slot, then select the destination SD Card in the second slot.

6. Now select to start cloning the SD do not do anything with the Raspberry Pi until the cloning process is completed. Once completed, the newly cloned SD Card can be removed.

4   References

Latex Syntax Information

 http://www.docs.is.ed.ac.uk/skills/documents/3722/3722-2014.pdf

How to check the temperature of a Raspberry Pi

 https://www.raspberrypi.org/forums/viewtopic.php?t=34994

Various ways to secure a Raspberry Pi

 https://www.raspberrypi.org/documentation/configuration/security.md

How to setup hosts.allow and hosts.deny files.

 https://www.thegeekdiary.com/understanding-tcp-wrappers-in-linux/

How to setup the realVNC server on a Raspberry Pi

 https://www.raspberrypi.org/documentation/remote-access/vnc/

How to install xrdp server on a Raspberry Pi

 https://www.maketecheasier.com/enabling-remote-desktop-access-on-raspberry-pi/

How to check if a program uses TCP Wrappers

 https://www.tecmint.com/secure-linux-tcp-wrappers-hosts-allow-deny-restrict-access/

Common commands to remember for a Linux system.

 http://www.penguintutor.com/raspberrypi/useful-command-reference

How to find the configuration files of an Apache Web Server

 https://www.digitalocean.com/community/tutorials/how-to-configure-the-apache-web-server-on-an-ubuntu-or-debian-vps

How to install and setup Apache Web Server on a Raspberry Pi

https://www.raspberrypi.org/documentation/remote-access/web-server/apache.md

How to setup a cronjob with Rsync

https://www.vultr.com/docs/setup-file-mirroring-using-rsync-in-debian-ubuntu