

Installing & configuring



Learn how to install Arch Linux, a barebones rolling LINUX distribution.

Many people think of Linux as an operating system, but in fact it's actually just a kernel, the base. To make it into a proper operating system, you need to add a little bit more. As Linux is free and open-source, many people have done this, and each one has done it slightly differently, creating many different 'distributions', or 'distros' for short.

The main Raspberry Pi distribution offered at <http://www.raspberrypi.org/downloads> is Raspbian, a version of Debian. However, if you scroll down a bit more, you'll see some others, including one called Arch Linux.

So what's the difference between Raspbian and Arch? The main difference is the package managers and the way updates are managed.

Debian, and therefore Raspbian, are very strict on package updates. They have to go through testing, so the maintainers can be sure they are stable and work before releasing them to the regular users. This is good because it means software is almost guaranteed to work, but not so good as it means updates take a little while to reach users.

Arch Linux is different in this respect, releasing updates as soon as possible. For this reason it is called a 'rolling release' distro, since you only have to install it once and then whenever a package gets an update you can upgrade it there and then. This allows users to get updates more quickly, although it means software is more unstable. In case of trouble, you can simply image the SD card again.

The other major difference between the two is that Raspbian comes completely ready, while Arch comes with the bare essentials, allowing users to pick what they want to install. This makes setup harder for newcomers though, but this guide should help ease the process.

So, if continuous updates and your Pi the way you want it sound good, why not have a go at installing Arch? You will need an internet connection for your Pi though, so if your internet is particularly slow or you have a low download limit, it may be best to stick with Raspbian.

First download the latest image from

<http://goo.gl/az5T4> Then flash the image to the SD card, using Win32DiskImager or dd. More information can be found in Issue 2 of the MagPi. With that done, we can get on with the setup.

First boot

The first boot might take a little while longer, just wait until it's done. Once you get to the login screen, use the user name root and the password root.

You will then have a terminal open. You will notice if you try startx, it will not work. That gives you an idea of how little Arch comes with. Don't worry though, we'll get to installing a graphical user interface.

Before you begin doing anything, you may want to change the keyboard layout to your country. This is done by typing:

```
loadkeys country code
```

I'm in England, so my country code would be uk. A full list can be found here: <http://goo.gl/xZh9N>. This is only temporary, we will set it permanently later on.

Editing language settings

The default hostname for the system is alarmpi. I personally don't like this, and would rather it was something else. If you feel the same, you can change the hostname of the system by typing:

```
echo hostname > /etc/hostname
```

where hostname is the new hostname you want. I've used raspberrypi. It will not be effective until after a reboot.

Next we will change the language and the timezone. To look at the available timezone areas, type:

```
ls /usr/share/zoneinfo
```

Choose which area suits you best (for me it would be Europe) and then type:

```
ls /usr/share/zoneinfo/area
```

to see the specific timezones. Pick one, and type:

```
ln -s /usr/share/zoneinfo/area/timezone  
/etc/localtime
```

all on one line. My choice was London, so area would be Europe and timezone London. If you get an error saying "File exists", type:

```
rm /etc/localtime
```

Then type the previous command again.

Now to edit the locale. This is used to determine the country you live in so things like dates can be displayed correctly. To do this, we need to edit some files. Type the following:

```
nano /etc/locale.gen
```

find your country and remove the '#' symbol in front of it. For example, mine would be en_GB. When you are done, use Ctrl+O to save and Ctrl+X to exit. Then type:

```
locale-gen
```

Now, we need to make another file, so type:

```
nano /etc/locale.conf
```

and edit it to the same language and country code as before.

Finally we need to set the console's keymap so it fits the country's keyboard all the time. For this type,

```
nano /etc/vconsole.conf
```

and change KEYMAP to the country code you used with the loadkeys command previously.

All the language settings are set now, so if you want you can reboot to see the changes, by typing:

```
reboot
```

Using pacman

Arch Linux's package manager is called pacman, and we use that to install packages. To install a package, type:

```
pacman -S <package name>
```

Try it with the package sudo, because we'll need it later.

As Arch is a 'rolling release', quite a lot of updates have come out since the image was released, so to upgrade all your packages type:

```
pacman -Syu
```

These should both work fine straight away with the most recent image, though you need an internet connection. Because of how quickly updates come, it's recommended you run a full upgrade regularly, once a week or maybe even once a day to keep on top of it all.

Should you want to remove a package, you can do that with:

```
pacman -R <package name>
```

And to see a list of installed packages, type:

```
pacman -Q
```

Adding a new user

It is vitally important that we make a new user for our system, as logging in as root has security issues. To add a new user, enter:

```
adduser
```

and follow the instructions to add a new user.

Next, we need to add them to the sudoers list so they can still install programs, but in a more secure way. If you haven't already, install sudo. To add the user to the sudoers file, type:

```
export EDITOR = nano && visudo
```

This will allow you to edit the sudoers file with the familiar nano editor. Find the line that says root ALL=(ALL) ALL and copy it onto a different line, replacing root with the user name of the new user. If you would like to have it so sudo does not ask for your password, like on the default Raspbian install, put NOPASSWD: in front of the final ALL.

Finally, we can change the password for the root account, using the command:

```
passwd root
```

Then be sure to pick a secure password.

And with that, we're done with the basic setup! Type:

```
logout
```

to log out of root and login as the new user you set up.

Setting up a graphical user interface

This final part is optional, but if you'd prefer more than just a command line you should do it. To install a graphical interface, simply type:

```
pacman -S gamin dbus xorg-server xorg-xinit xorg-server-utils mesa xf86-video-fbdev xf86-video-vesa xfce4
```

all on one line. Once the installation is finished, type:

```
cp /etc/skel/.xinitrc ~/.xinitrc
```

and then:

```
echo "exec startxfce4" >> ~/.xinitrc
```

and finally:

```
startx
```

Your graphical environment should then start. Congrats, you now have a working Arch Linux system on your Raspberry Pi! You may want to edit the config.txt, but this process is the same as Raspbian. Have fun!

Article by Alex Kerr