GET HELP FROM THE COMMAND LINE

The basic commands you need to get advice from the Linux command line

On our beginner's guide to the command line (*The MagPi* #54), we looked briefly at 'man', the manual you can access from the command line.

The man tool is so important that we think it deserves a more thorough explanation. And man isn't alone in offering help on the command line. Other commands like **whatis**, **info**, and **apropos** all offer support and assistance. And let's face it, support and assistance are what you will often need at the command line.

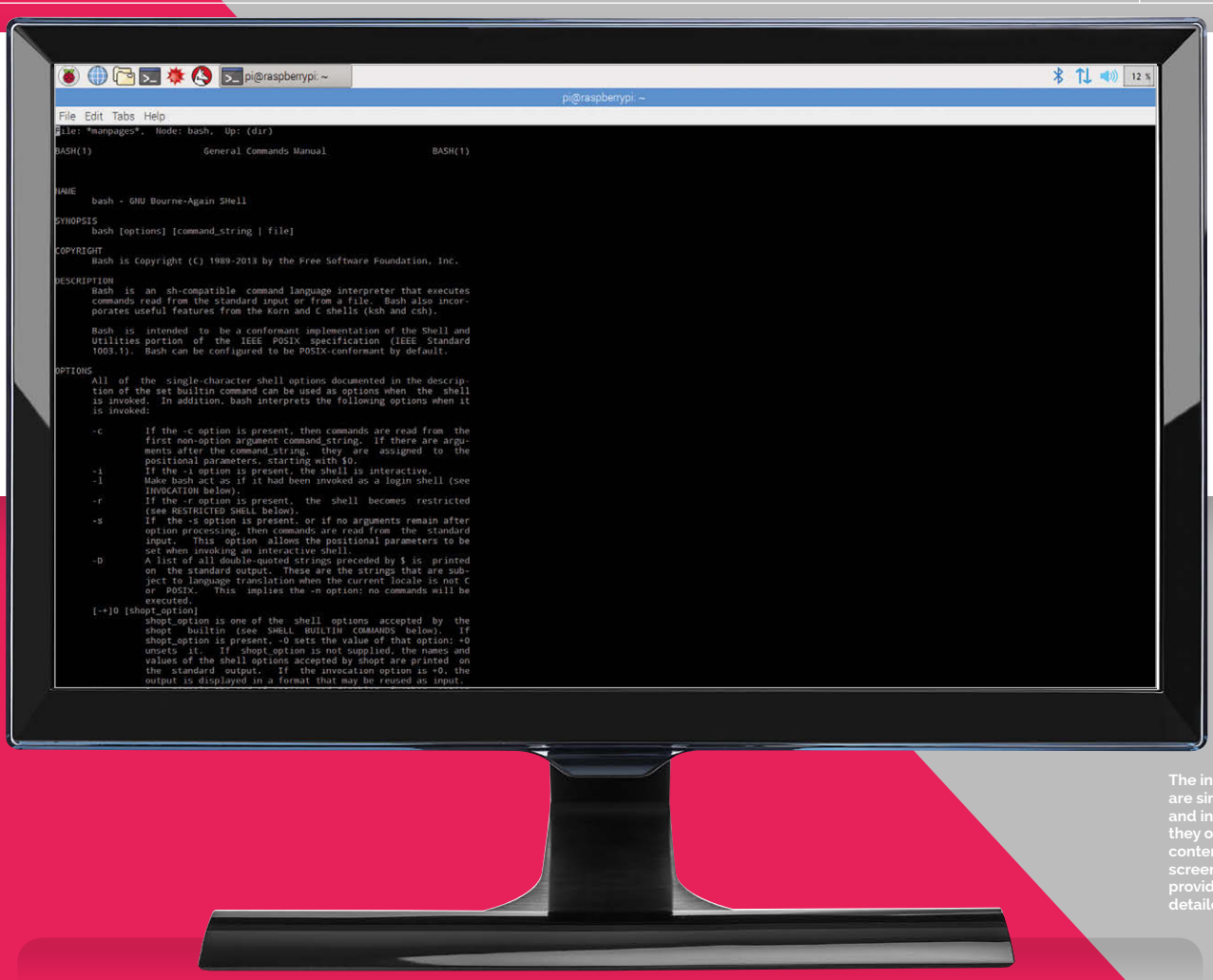
Even seasoned coders don't always know the correct command to type into the Linux terminal. This guide is all about the various ways to get help at the command line, so no matter what command you come across, you'll be able to find out more information on how to use it.

man

Your first point of call for getting help on the command line is man (short for 'manual'). Enter **man** followed by the name of a command to get detailed information about it. For instance, enter:

```
man passwd
```

...and you will see detailed information about the tool used to change your password. Man screens are displayed one page at a time. Press the **SPACE** bar to



The info screens are similar to man, and in many cases, they offer the same content. But some info screens (like bash) provide much more detailed content

move to the next page, and press **Q** to exit the page and return to the command prompt.

Man pages can be a bit tricky to read at first, but you'll soon get the hang of it.

At the top are the Name, Synopsis, and Description sections. Read these to get an overview of the command. Below them you'll find options and parameters; read these carefully to discover ways to expand your usage of each command. It's a good idea to use man on any commands you know, and read the manual for any new Linux commands you come across.

You can even read a man page for man:

```
man man
```

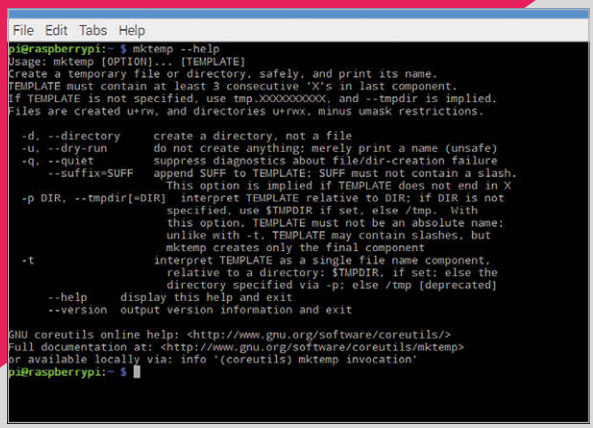
Press **H** in the man screen to view a summary of navigational key presses. These are worth learning so you can do more than press space to move to the next page.

Man's lesser-known partner is 'info', which is used to display information pages associated with commands. Sometimes these are the same as the man pages. In other cases they provide a different description. Try these:

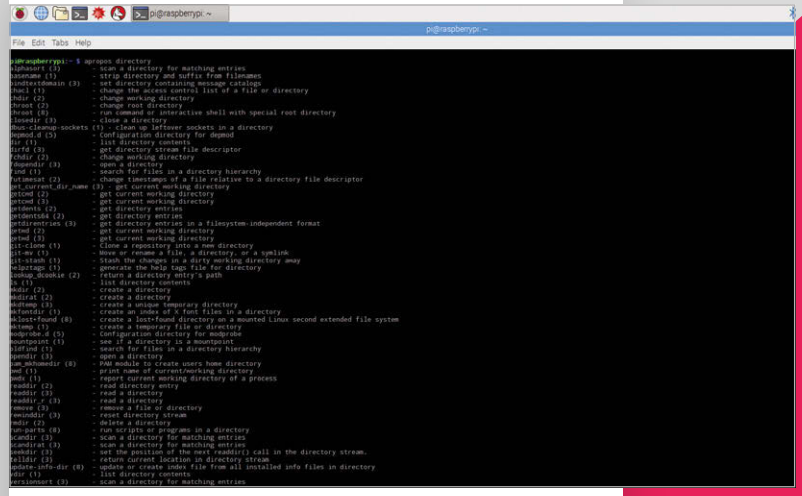
```
man bash
info bash
```

While **man bash** gives you a brief description of the GNU Bourne-Again Shell and the options used with the bash command; **info bash** gives the whole history and hundreds of pages of detailed information.

Press **H** on an info screen to view the controls for navigating such long documents. As well as **SPACE** to move down, you use **DELETE** to go back a screen, **TAB** to highlight links, and **RETURN** to use them. Press **Q** to exit the help screen.



Many commands feature a built-in help option, accessed with -h or --help. Using it offers a brief outline



Apropos is used to search the manual for matching words. With it, you can find commands based on subjects, such as directory, password, or links

FINDING COMMANDS

As you become more familiar with man and info, you'll start searching for commands to look up. Here, the **man -k** command comes in useful. In particular, try this:

```
man -k directory | more
```

This command lists all available man entries. Press **SPACE** to run through them one at a time. The **man -k** option is worth remembering. If you use **man man**, it tells you the **-k** option is 'equivalent to apropos'. Apropos is used to search manual page names and descriptions. It's a handy way to find commands when you don't know their names.

For instance, enter:

```
apropos directory
```

...and you'll get a list of all the commands that have the word 'directory' in their description or page name. Here you'll find common commands such as **ls**, **cd**, and **pwd**, but you'll also find less obvious commands, such as **mktemp**.

Next to each command is a number, like (1) or (2). These correspond to the section numbers of the manual (view using **man man**).

The section numbers are useful for guiding you to the commands that can be used on the command line. As a general rule, **1:** Executable programs or shell commands, and **2:** System calls, both tend to be worth investigating. Higher numbers are for library calls, special files, and kernel routines for advanced users.

You can find out more information about any command using man:

```
man mktemp
```

This command gives you detailed information on how to create temporary directories.

TAB AUTOCOMPLETE

Another way to find files is to use 'tab autocomplete'. By pressing the **TAB** key, you can automatically complete commands, files, and directories on the command line. If you're not doing so already, learn to press **TAB** a lot on the command line: it's a good way to discover new commands.

Take the **apt** tool, for example. There are **apt-get** and **apt-cache**, but did you know about **apt-config** and **apt-key**?

Enter:

```
apt
```

And press the **TAB** key twice. It will display all the different types of apt available.

You can even run through the letters of the alphabet. Enter the letter 'a':

```
a
```

And press **TAB** twice to view all the commands beginning with 'a'. You can then use man to look up commands. It's a great way to broaden your knowledge of the command line.

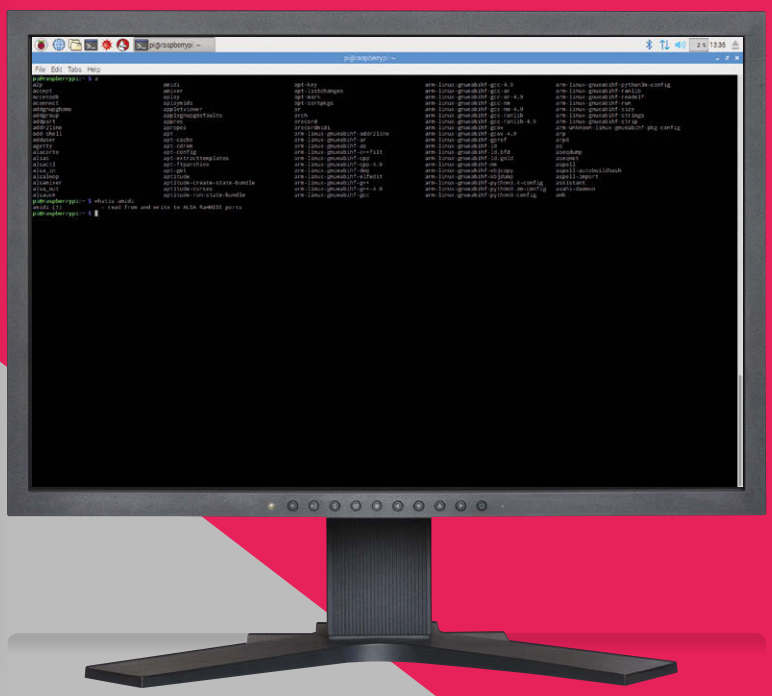
EXPRESS HELP

Many commands also offer a help feature as an option. Help is typically accessed using **-h** or **--help**:

```
mktemp --help
```

Enter a single letter and press TAB twice to view all the commands available in that letter. Then use whatis to get a short one-line description of each command

This command displays the options offered by the mktemp command. It's the same as the first page of mktemp's man file, but saves you digging in and out of the full document.



Not all commands make use of **--help**. Some, like **ls --help**, display the full man document (you can pipe this through **less**:

```
ls --help | less
```

...but it's typically easier to use **man ls**. Some commands don't implement the help option at all.

```
pwd --help
```

...returns 'invalid option'. But it's worth trying when you are experimenting with new commands.

One final command worth using when searching for commands is whatis:

```
whatis pwd
```

This example returns 'print name of current/working directory'. Often, this brief description is enough to let you know what it does, or at least tell you if it's something you'd like to investigate further with man or info.

These are just some of the tools you can use to get help at the command line in Linux. While the command line may seem intimidating at first, you're far from alone in this text-only environment.

WEB SEARCH

One of the advantages when using a desktop interface, like Raspbian, is that a web browser – and a search engine – is just a click away.

Getting online from the command line is a lot easier than you'd imagine. There are many different text-based web browsers that enable you to access Google, Bing, DuckDuckGo, and other websites without having to boot into the PIXEL desktop interface.

We're going to use:

```
sudo apt-get update && sudo apt-get upgrade
sudo apt-get install elinks
```

Now you can open the web browser from the command line using:

```
elinks
```

The elinks interface is full-screen, so it replaces the command line. Press **g** to open a URL field. You can enter full URLs, such **http://www.google.com** or just shortened versions, such as **raspberrypi.com**.

Better yet, there are a few key bindings for helpful sites. Press **g** then enter these shortcuts:

- **d** – dict.org search
- **sd** – Slashdot
- **g** – Google search

You can also enter Google search terms in the URL field. Press **g**, then enter 'g the magpi' to search for our website in Google.

Other keyboard shortcuts can be used to navigate the program:

- **g** – Goto URL
- **Down Arrow** – Next link
- **Up Arrow** – Previous link
- **Return** – Select link
- **Left Arrow** – Back
- **u** – Forward
- **q** – Quit
- **.** – Toggle link numbering
- **%** – Toggle colours
- **t** – New tab
- **T** – Open link in new tab
- **>** – Next tab
- **<** – Previous tab
- **c** – Close tab