

LINUX FORMAT

Get into Linux today!

70 Pages of
tutorials
and features

- » Voice control your Pi Zero
- » Dual-boot Linux & Windows
- » Improve laptop battery life



Coding Academy: Fire up the micro:bit with MicroPython

ESCAPE GOOGLE!

Take back control!

Build your own cloud with
the best open source tools



Peerless storage

“The issue of simultaneity becomes a real problem. It turns out that actually Einstein was right”

Allen Samuels creator of Ceph distributed storage

Stay hidden from GCHQ

Keep safe online with
next-gen Tor browsing

Servers
Discover CentOS

» Get to grips with the best
server distro and our guide



Roundup
Explore BSD

» The best options for the
'other' open source kernel



Future



3G/4G
hotspot



Beamforming
technology



Downloads &
multimedia hub

Synology Router RT1900ac



Install DS router to find and complete
setup using your Android or iOS device.

YOUR NETWORK, SMART AND SWIFT

Designed for homes and offices, Synology Router RT1900ac broadcasts both 2.4 and 5 GHz spectrums for combined data transfer speeds of up to 1900 Mbps. Fine-tune traffic flow with advanced network tools like Application Layer QoS and Parental Controls. Do more with full-fledged Synology software packages, and get a revolutionary user experience with Synology Router Manager (SRM).

What we do

» We support the open source community by providing a resource of information, and a forum for debate.

» We help all readers get more from Linux with our tutorials section – we've something for everyone!

» We license all the source code we print in our tutorials section under the GNU GPL v3.

» We give you the most accurate, unbiased and up-to-date information on all things Linux.



Who we are

This issue we asked our experts: We're replacing proprietary online services with open source options. What's your favourite or most essential open source service?



Jonni Bidwell

I really appreciate when websites make all their code open, like *Gaming on Linux* has just done (thanks Liam). But I've been using *ownCloud/Nextcloud* pretty much since being in this job. And it's never let me down once. The Dropbox client used to be terrible, maybe it's better now, but that point is moot.



Nate Drake

I recently switched from backing up my files with Dropbox to using *Nextcloud* on a home server. [Ed—we swear we aren't taking backhanders from *Nextcloud*.] The data never leaves your place and you can verify it's encrypted server side. You can also store terabytes of data, all for the cost of a Raspberry Pi and an external drive!



Adam Oxford

I'm migrating eight years of Evernote to SimpleNote. I've been a premium subscriber to Evernote since 2009, and it's been an invaluable. But the demise of the *Everpad* client means there's no good offline client for Linux—which is great as it's encouraged me to switch to an open source alternative.



Les Pounder

My blog [shameless plug <http://bigl.es>] is hosted on Ghost, a low resource open source blogging platform. Rather than a bloated Wordpress blog that requires frequent patching. I prefer Ghost as I can quickly write posts in markdown, incorporate HTML where I need it and quickly create posts that are full of media.



Mayank Sharma

I don't believe any server-centric tutorial is complete without a word on *PageKite*. The pay-what-you-want tunneling solution makes local websites and services publicly accessible immediately without any fuss. It's built using open source tools and ensures privacy and security by using SSL and not exposing your IP address.



Slippery slope

» In the USA there's the Fourth Amendment, here in Europe there's the European Convention on Human Rights and both protect an individual's rights when it comes to government interference over person, possessions and private life. In the past that largely meant the government couldn't just waltz into your property and seize any documents (or persons) it felt like, listen in on your private conversations or stop you from moving around the country without a damn good reason. Sounds reasonable, right? It seems in the internet age those rights no longer apply.

This isn't tinfoil hat stuff, it's happening right now. In the UK the new Investigatory Powers Act – that gives many governmental agency the power to freely view your internet history – has come into place. Meanwhile in the USA since the Patriot act and Snowden revelations, it's apparent most US communications can be tapped without warrant. Makes you feel all warm and fuzzy inside, right?

A huge problem is the lackadaisical attitude of many to such intrusion. The mantra of 'nothing to hide, nothing to fear' entirely misses the dangers that government and humans are entirely fallible and prone to extreme change. While you might feel your government is entirely loving and friendly today, you've no control over how it might be shaped 10 years from now or the general abuse of these powers by individuals. That's beside the fact these powers are often (ab)used to curb anti-government movements, social change and protests. While at the heart of it all, not wanting to be searched for no reason in no way implies you're guilty of anything, nor should it.

So this issue we're looking at how we can reclaim ownership of our cloud, escaping Google and 'the man' at the same time. Taking ownership of your many online services, using open alternatives and ensuring your data is secure while using them.

We also delve into the Invisible Internet Project, a post-Tor service, explore the BSD kernel, improve your laptop experience, explain basic terminal sysadmin tools, dual boot Linux and so much more we've run out of room—enjoy!

Neil Mohr Editor

» neil.mohr@futurenet.com

Neil

Subscribe & save!
On digital and print, see p26

Contents

"I don't love studying. I hate studying. I like learning. Learning is beautiful." – Natalie Portman

Reviews

Intel Core i7 7700K.....15

Oh Kaby Lake how deep your waters run, with SSE4 and GPU fun. Should we upgrade or wait for Ryzen? Let us find out, so that concludes our little rhyme, oh what fun.



» A new CPU has been released, is it good enough to tempt you to upgrade?

Nvidia GTX 1050 Ti.....16

Linus might stick two fingers up at Nvidia, but there's no denying the quality of their GPUs or is there? We test out the budget offering to see if it's good enough to buy.

Synology DS916+.....17

A new generation of NAS devices are pushing the limits of what that actually means. This mid-range model from Synology even delivers *Docker* support.

Total War: WarHammer..18

Total War comes to the WarHammer universe, well, it hadn't really ever left but the ultra-realistic gaming strategy series takes a pop at the fantasy world of orcs and elves.

Rainbow HAT.....57

Les Pounder tests an Internet of Things board designed for use with the new Android Things platform, but finds it's just as handy when it comes to Python-powered projects.



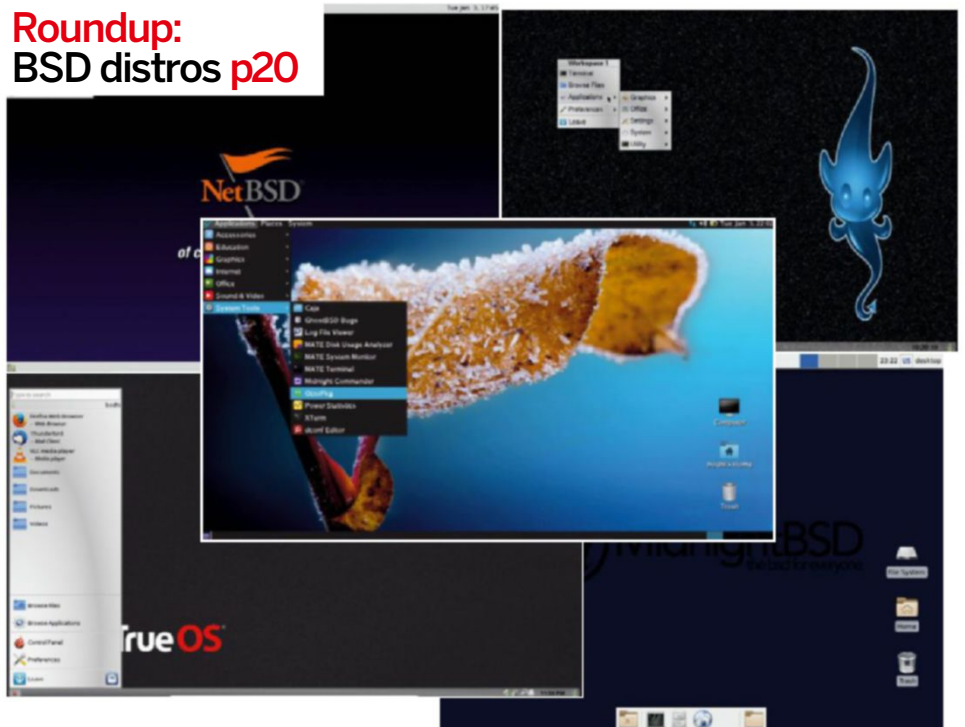
» It's an unassuming board that offers multiple methods of input and output.

Escape Google! Own your cloud

We take the best open source tools and services and build our own impenetrable castle in the cloud. Do it now on p28.



Roundup: BSD distros p20



Interview



“Tape drives never went away. Spinning rust is never going to go away either.”

Allen Samuels on the open source project Ceph p38

EXCLUSIVE
Ubuntu Remix
SPECIAL EDITION

64-bit

» Super-stable Kernel 4.8
» 6 desktops for you to try!
» Powerful application suite

» **32-bit heroes**
Two feature-packed, low-resource distros. Perfect for older PCs!

LINUX LIVE DISC: READY TO RUN
EVERYTHING YOU NEED TO GET STARTED WITH LINUX

Siduction 16 Porteus 3.2

On your FREE DVD

Ubuntu Remix 16.10 64-bit

Siduction 16.1 XFCE 32-bit

Porteus 3.2 Cinnamon 32-bit

» Only the best distros every month
» Plus HotPicks, code and library **p96**

LINUX **FORMAT** **67** Pages of tutorials and features

FREE DVD! **OPENSUSE LEAP 42.2**

YOUR OPEN SOURCE SMART HOME

LINUX **FORMAT** **67** Pages of tutorials and features

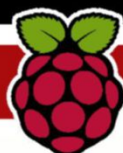
YOUR OPEN SOURCE SMART HOME

LINUX **FORMAT** **67** Pages of tutorials and features

YOUR OPEN SOURCE SMART HOME

Subscribe & save! p26

Raspberry Pi User



Pi news..... 56

The Foundation [the Asimov one?—Ed] releases its Maker curriculum and Compute 3.

Rainbow HAT 57

An unassuming board that offers multiple methods of input and output.

Raspberry Pi GPIO..... 58

Les Pounder explains how to use GPIO Zero to create a simple 1980s-inspired memory game.

Pi voice..... 60

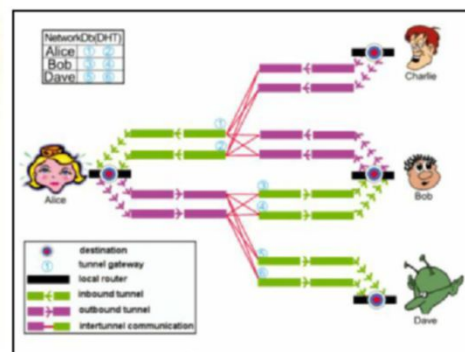
Nate Drake hates germs bigly style, so he's created a voice controlled Pi so his tiny hands don't have to touch the filthy thing ever again.



In-depth...

Invisible Internet Project 42

Disappear into the I2P with **Nate Drake** as he introduces the new Tor+ ultra-secure system.



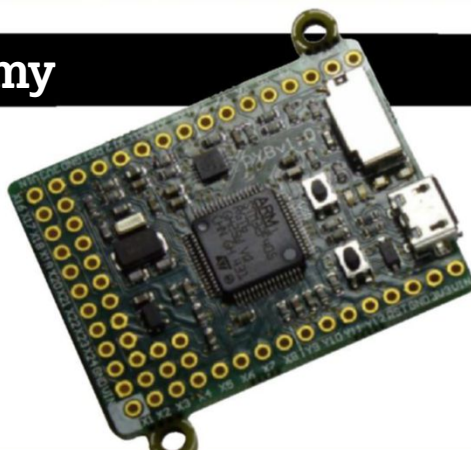
Coding Academy

psutil for sysadmins 84

Mihalis Tsoukalos explains how to use the cross-platform psutil module to examine your Linux system and become a power user and happy system administrator.

MicroPython 88

Les Pounder introduces us to MicroPython, a leaner implementation of Python 3 for microcontrollers and how to use it on the now widely released BBC micro:bit.



Tutorials

Terminal basics

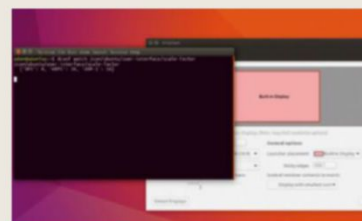
Sysadmin skills 66

Discover the essential commands and skills with **Adam Oxford** that will turn you into a terminal system admin.

Laptops

Going mobile 70

Ensure your battery keeps on ticking with **Adam Oxford's** help as he fires up his trusty Dell XPS 13 to offer a few hints and tips.



» Linux loves being mobile.

Linux gaming

Gamepad control 74

If you want to make the most of your games you'll need control options – and a gamepad could be just what you need.

Grub explained

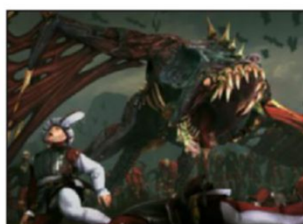
Dual-boot Linux 76

Get more than one OS living on your hard drive, we're avoiding that 'W' word...

Servers

Hello CentOS 80

Get to grips with the server distro that is best buddies with the enterprise loved Red Hat.



» Our subscription team is waiting for your call.

Regulars at a glance

News..... 6

Dell sells a bigly amount of Linux laptops, *Mir* looks to hit v1.0 in 2017, Librecore is launched and *Wine* 2.0 ushers in a new age of releases.

Mailserver..... 10

A new slim-line mail section as you lot aren't writing enough letters, lazy and low energy! No fake letters here.

User groups..... 13

Les Pounder goes back to his and the Pi's JAM root in sunny Preston.

Roundup 20

Splitter **Mayank Sharma** is using that other kernel called BSD, if you fancy a try we look at the best options.

Subscriptions 28

We've got incredible deals, the best deals, yuge deals, you won't believe how beautiful our subs deals are!

Sysadmin..... 46

Mr. Brown's delves into the 'Datacentre Operating System' a contender for running containers as well as big data applications.

HotPicks 50

Alexander Tolstoy is too busy to write any memos or record videos of hotel rooms, he's too busy recording golden FOSS like: *Seamonkey*, *Gnome Recipes*, *VPaint*, *Mermaid*, *Kadu*, *Inkscape*, *MPD*, *Eqonomize*, *Darktable*, *2H4U*, *Veraball*.

Overseas subs 65

We ship *Linux Format* all around the globe, even the USA, get a sub now!

Next month 98

The student surpasses the teacher. We explain how you can become a master at the alter of the black terminal, the ultimate interface.



THIS ISSUE: Laptop Linux love » Mir moves » Rebooted Libreboot » More Wine!

HOT CAKES

Dell's big Linux sales

Dell boasts of selling “tens of millions of dollars’ worth” of Ubuntu laptops – but don’t go expecting other distros any time soon.

In a recent video interview with Bryan Lunduke (which can be watched at <https://youtu.be/9EQfu4Glfhs>), Barton George, senior principal engineer, Office of the CTO at Dell, outlined the company’s commitment to selling laptops with Linux preinstalled. However, much like Henry Ford’s jokey colour options for the Model T, it appears that for the foreseeable future, you can choose any distro you like – as long as it’s Ubuntu.

Dell’s enthusiasm for machines with Linux pre-loaded makes sense, with George’s assertion that “from an initial investment of \$40,000 it has returned, over the four years, tens of millions [of dollars]”, though he wouldn’t be drawn on exact sales numbers.

The interview coincides with the launch of the Dell Precision 3520, a mobile workstation aimed at developers that can come with Ubuntu 16.04 LTS ready installed. It features a 15-inch display, and as with other Dell laptops, comes in a range of hardware configurations. At CES 2017 in Las Vegas, Dell unveiled the Precision 5720, too, a powerful all-in-one PC that can also come with Ubuntu preinstalled.

George also hinted at more Linux laptops, including the 5520, a “precision workstation”, which is available to buy now, as well as “two big ones coming up... if you thought the 5520 was big and beefy... we’ve got the 7720 and 7520, and those are really über-powerful”.

This will join the already-impressive list of Dell workstations that can be bought online with Ubuntu preinstalled, such as the Precision 15 3000 series. But what if you’re not a fan of Ubuntu? While Dell does sell servers certified for



a range of distros, including SUSE and Red Hat Enterprise Linux, you’ll only find Canonical’s offering on Dell’s desktops. According to George, “We have no plans in the foreseeable future to ship other distros on our hardware... To try and spread our [small team] across more distros would do more harm than it would good.” If you want

» **Moving to support Linux on an all-in-one PC is a bold move from Dell.**

other distros were supported. At least Dell is now offering Ubuntu 16.04 LTS, which is supported until 2021.

We talked to Barton George ourselves in **LXF220**, and he promised us that, “[Dell’s] overall goal is to continue to provide a first-class Linux-based developer laptop.” When asked by Lunduke why Dell continues to offer Linux laptops when its competitors, such as Asus and Acer, have stopped after brief dalliances, George remarked that, “We’ve been doing it for a long time, so it’s not just that we started with Project Sputnik,” which we discuss in **LXF220**. George continues, “The difference with Sputnik is taking Linux and putting it on a high-end laptop, which is something that was a new position to folks here.”

We’ve always been big supporters of Dell’s decision to put Linux on its powerful workstation laptops, rather than relegate it to its cheaper, and less powerful, machines, like so many of its competitors, and it’s good to see that decision pay off with such impressive sales. Unfortunately, it doesn’t look like we’ll be able to walk into a high street shop and buy a Linux-toting Dell machine any time soon, but to see the

wide range of Ubuntu-powered devices sold by Dell, head over to its website at www.dell.co.uk and choose Ubuntu as the OS. Perhaps the best

news is that by choosing Ubuntu on applicable laptops, it cuts the price by around £110, because you’re no longer buying a Windows 10 licence as well. A pretty nice trade-off if you ask us.

“Choosing Ubuntu on a laptop cuts the price by around £110, as you’re not buying Windows 10.”

another distro on your Dell machine, you can still choose to have it shipped without an OS, though George was hopeful that Dell would continue to push fixes and drivers to make sure

UBUNTU

Canonical promises Mir

It's coming, honest – version 1.0 is set for 2017.

In a guest post for the Ubuntu website (<https://insights.ubuntu.com/2017/01/17/mir-2016-end-of-year-review/>), Alan Griffiths, a software engineer at Canonical, promised that 2017 would see the release of Mir 1.0, the display server developed by Canonical to replace X Windows System in Ubuntu.

While it has only been deployed on mobile devices running Ubuntu Touch in the past,

Griffiths promises we'll start seeing it in future releases of Ubuntu for desktops, along with support for the Vulkan API.

According to Griffiths' post, "2017 will see a cleanup of our 'toolkit' API and better support for 'platform' plugin modules. We will then be working on upstreaming our mesa patch. That will allow us to release our (currently experimental) Vulkan support. We've also been working on reducing latency but the big wins didn't quite make the end of 2016."

With Mir 1.0 powering the Unity 8 display server, and upstreaming Mir support into *GTK 3*, *Qt*, *SDL2* and *Kodi*, we may finally see the elusive new technology this year in the milestone 1.0 version. Canonical has yet to set a date any more concrete than "2017", but with the imminent arrival of Ubuntu 17.04, and work underway on Ubuntu 17.10, could late spring be Mir's moment to shine? We certainly hope so.



➤ Could 2017 finally see the release of Mir 1.0 on various Ubuntu machines?

COREBOOT

Librecore announced

A sensible alternative to Libreboot, but can it mend the divide?

Libreboot (see *Tutorials LXF214*) started life three years ago with a lofty ambition to replace proprietary BIOS firmware, and to be a distribution of Coreboot without binary blobs. Unfortunately, the good work the project has achieved has often been overshadowed by the antics of its head developer and original author, Leah Rowe. This includes her calling for a boycott of the Free Software Foundation only a year after it was endorsed by the organisation, and a few months after it officially became part of the GNU Project, due to alleged mistreatment by the FSF (which it denies) of an employee.

She then accused FSF of "not letting Libreboot go", and according to Libreboot contributor Damien Zammit, she had decided to separate from GNU unilaterally, while forwarding her personal views as those of the Libreboot community. Rowe also had a public spat with



➤ Daft rabbit, proprietary BIOS firmware is for kids.

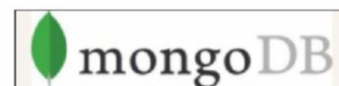
Timothy Pearson of Raptor Engineering, who claims that Rowe has refused to pay him for contract work (which Rowe denies, as reported by Phoronix at <http://bit.ly/LXF221boot>). Librecore, which includes contributions from Zammit, is a new Coreboot downstream that aims to move on from the community clashes caused by Rowe, while updating its firmware and build system. For more information, and links to its GitHub page, visit the project's website at: <http://librecore.info>.

Newsbytes

➤ In the first big release as part of its new update processes, *Wine 2.0* has now been released. The latest version of the software that allows Windows apps (and even intensive games such as the new *Doom*) to run in Linux. The new version bump brings a ton of new features including Direct3D 10/11 support, Office 2013 support, plus a ton of bug fixes, including a number for playing games, and includes better support for Windows apps such as *Visual Web Developer Express 2008*, *Battle.net* launcher and *SMARegistry Backup* installer. See www.winehq.org.

➤ *PulseAudio 10.0* has been released, with improved VoIP capabilities. It allows the Bluetooth profile to automatically switch when using VoIP apps with a Bluetooth headset – it now switches from the A2DP profile (used for listening to music) to HSP, which is suitable for telephony. The profile then switches back to the A2DP profile once the call ends. *PulseAudio* is recommended for Linux users who want to do more with the audio on their machine. To find out more, check out the changes log at www.freedesktop.org/wiki/Software/PulseAudio/Notes/10.0.

➤ In January it was revealed that around 28,000 (at the time of writing) MongoDB databases have been hit by hackers, having their content removed and replaced by a ransom note. While it started off as individual attacks, it was reported that these intensified, with multiple groups joining the initial hacker. The ransom fees range from \$150 to \$500, a relatively small amount that was chosen in order to encourage the victims to pay to get their data back. Perhaps worst of all, it emerged that some groups who were paid the ransom were not the initial hackers, forcing victims to pay twice, while other groups did not back up the data they removed, leaving victims to pay for nothing. The scale and speed of the attacks led some people to term it a 'gold rush'. If you're concerned about being a victim of these attacks, MongoDB has published an updated guide on how to protect yourself at www.mongodb.com/blog/post/how-to-avoid-a-malicious-attack-that-ransoms-your-data.



➤ A 'gold rush' for hackers brought down 28,000 MongoDB servers.

Comment

Coping with data explosion

Clyde Seepersad



We are living in an age when data is generated at

unprecedented volumes and speeds, particularly in the corporate world. According to industry estimates, data is expanding at a rate of at least 50% every year, and it currently comprises 80% of all corporate assets.

Not too long ago, all data would have been stored in conventional data centres – physical, onsite hardware facilities that were managed with little automation. However, their many inefficiencies, coupled with their increasing ineffectiveness in dealing with the burgeoning amount of data being produced, made it imperative to search for a viable alternative.

Enter the cloud – an off-premise form of computing, where data storage takes place over the internet. According to IDG's 2016 Enterprise Cloud Computing Survey, the average company runs 45% of its operations in the cloud, a figure that will only grow.

OpenStack has emerged as a leading cloud solution. In fact, the 2016 Open Source Jobs Report, published by The Linux Foundation and Dice, revealed that 51% of hiring managers say knowledge of OpenStack and CloudStack are driving open-source hiring decisions.

Not only does OpenStack offer seamless management and scalability of private and public clouds, but its adoption by huge corporations such as eBay, PayPal and IBM is making it a de facto standard. The fact that it is open source offers users greater flexibility, as they're not tied down by vendor commitments.

There's no time like the present to get in the know. Download a sample chapter from The Linux Foundation's course The Essentials of OpenStack Administration, which will guide you in creating and managing private and public clouds with OpenStack.

<http://go.linuxfoundation.org/lfs252-sample-open-stack-admin-fundamentals>



› Clyde Seepersad is the general manager of Training at The Linux Foundation.



Distro watch

What's behind the free software sofa?

BITKEY 14.1.0

The first version of BitKey, a Debian-based distro aimed at Bitcoin users, is now available to download. It brings tools for secure air-gapped Bitcoin transactions, and includes *warpwallet*, *coinbin* and *libbitcoin-explorer*. You don't need to install the distro – it comes as a live CD and is stored in RAM, and uses the TurnKey GNU/Linux build system to create a



› Everything you need for ultra-secure Bitcoin transactions.

secure live CD with all the tools even the most paranoid Bitcoin user needs. Find out more and download the ISO at <https://bitkey.io>.

ARYALINUX 2017

AryaLinux 2017 is a source-based distro that's been built with Linux From Scratch and uses a source/ports-style package management, with a custom package manager called *alps*. The latest version is 64-bit only and comes with package updates and a new set of scripts to help you build it from scratch. It supports KDE Plasma and Gnome desktop environments, though a graphical front-end to *alps* missed out on the release; it will be released

as a package later on. For more info, head to <http://aryalinux.org/release-notes/aryalinux-2017.php>.



› Arya Linux lets you build it yourself from scratch, and comes with some interesting features.

PARTED MAGIC 2017 01 08

The latest version of this live CD distro that focuses on disk partitions and data recovery has been released. It uses X.Org Server 1.19.0 and the kernel has been upgraded to Linux 4.9.1, while compatibility for a range of graphics cards has been improved. According to the announcement, "We also added a few programs and made a few minor nitpicks most

people didn't even notice." See for yourself (if you pay \$9) at <https://distrowatch.com/qxjxuywhvada>.



› Conjure up control over your hard disk.

KAOS 2017.01

2017's first KaOS ISO image has a new look, with a new sddm theme that uses a layered QML mode. You can switch between KDE Plasma and Wayland, and as it's a rolling distribution, you get cutting-edge packages. Linux 4.8.15 includes a change to how the kernel image is created, so now it is triggered for more instances, like filesystem and Systemd updates, so you avoid mismatches with the Systemd

version on updates without having to manually update the kernel. For more information, visit <https://kaosx.us/news/2017/kaos01>.



› KaOS has a brand new look for 2017 – and it's rather fetching!

NEW

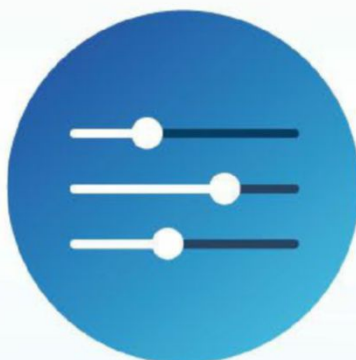


CloudNX

next generation Cloud Servers



**UK data centres,
UK support**



Flexible



**Pay as you
use pricing**

Enterprise features

- Load balancing
- VPN
- External firewall

Set up in seconds

- Easy to use
control panel

Cloud Servers from:

£15.00

per month ex VAT charged at 20%

Call **0808 1686 777**

or visit **fasthosts.co.uk/cloud-servers**

SERVICES • WEB HOSTING • DOMAIN NAMES • EXCHANGE EMAIL



Mailserver

Write to us at *Linux Format*, Future Publishing, Quay House, The Ambury, Bath BA1 1UA or lxformat@futurenet.com.

» Pi of Things

I read with interest the Super Pi article, and in particular the IoT part. I have been looking for some time at using a Pi as a hub for IoT devices, instead of proprietary systems that come with their own hub. In theory, you could end up with several hubs, and all current systems seem to have dubious security standards at best.

I'm also keen to make some of my own IoT devices. To that end, I have discovered and purchased from a talented guy in the US, who has developed, makes and sells an Arduino clone called a Moteino, that is very low power and, at its smallest form, only 3 x 2.2cm,

with a radio tx and rx attached using the SMI band.

He has also developed a secure gateway using a Pi to talk to any device with his Moteino, and it also has the ability to talk to other devices. All the details are available on his site <https://lowpowerlab.com>. As an added bonus, he also has a number of purpose-made Motion devices and other bits, one of them being for cleanly shutting down the Pi at the touch of a button, but also, at the loss of power, as he's added battery backup for power outages.

Paul Farrar, via email

Neil says: We hope you read our Secure Smart Home feature in LXF220, where we covered some

of the points you made – though, as with security tools, there are more IoT and maker products out there than we have time to cover – about the need to keep control of all the systems in your networked home, from IoT devices to the systems that control them.

» Digital DVDs

I've just started a subscription to Magzter Gold, and am reading your magazine, *Linux Format*, through its app on my tablet. I was hoping to be able to download the DVD but, after viewing your website, I didn't see an obvious way to do so. I did see the instructions in the 'Subscriber Area' page on the website, but these instructions don't seem to be applicable to Magzter Gold subscribers. Are Magzter Gold subscribers able to get access to the DVD downloads, and if so, how is that done?

Steve Walsh, via email

Neil says: I'm sure that's totally legit – who knows, these days? To be honest, as we're peddling, erm, packaging open-source software, we do make the ISO freely available via torrents on our



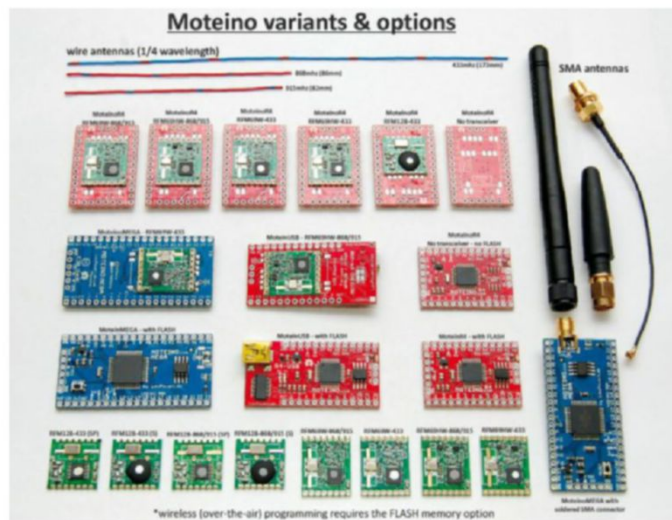
» The archives are there for subscribers to access and to download the DVD ISOs.

archive website. Take a look at www.linuxformat.com/archives – even if you're not a subscriber, you can get hold of the code packs and discs, and we also make the odd article openly available, too, although there's no way to search for those PDFs.

» Error corrected

Great magazine, thank you. I've just re-read your article on the ultimate home server in LXF213, and realised you go into a lot of detail on RAID to protect data, but on the previous page, in the small hardware box, you say any dual-core Pentium will do.

I think this is a tad lacking in consistency, as the standard Pentiums *et al* do not support ECC, and if you look at the reliability of DRAM, as shown in some tests (www.cs.toronto).



» There's a world of maker devices out there we're yet to discover!



edu/~bianca/papers/sigmetrics09.pdf), you don't want to store data, in such a safe place as RAID, the already flipped bits your low-tech CPU has passed to the storage system from RAM.

Please recommend a system that provides ECC, because if you do not go Intel, you can have ECC for not a lot more than not using ECC, and thus protect all parts of the system. I know Intel leaves memory as the only unprotected element of a system to try to get people to pay for Xeon, but home users do not need Xeon, just ECC.

Chris Lee, via email

Jonni says: I definitely should have mentioned ECC RAM and its *raison d'être* in the Ultimate Home Server feature. However, I'm not sure I agree with you about ECC memory being absolutely necessary here. Definitely memory errors occur, and having ECC RAM will protect

you against most of them, but I don't think that peace of mind is really worth the platform upgrade. There are all kinds of other things that could go wrong, with the same or higher probabilities as a memory bitflip – disk errors, power failures, Steam crashing, and users messing up their files.

One case where you are quite correct to insist on ECC is if you have a large ZFS array. ZFS stores much more data in memory than other filesystems, so it's much more likely that a random in-memory bitflip is going to affect file data or metadata.

I know, the right bitflip might, for any filesystem, cause data to get written to the wrong sector of the drive and cause untold damage, but you'd have to be really unlucky for that to happen. If you have extra money to spend to allow ECC support, then one could argue that the money would be better spent on a UPS or more reputable memory/other



› We're still planning on running a home server series at some point...

components. If you really care about your data, you want redundancy (probably a four or six-drive RAID 6, maybe four-drive RAID 10), daily backups (on and off-site), a checksumming filesystem like ZFS or Btrfs, and limited access to both the files

and the hardware. All of this is probably overkill for a simple home server appliance. We even had someone write in and say the RAID 1 scheme I recommended was unnecessary and over-complicating the setup. I disagreed with them, too. **LXF**

Letter of the month

Secure stuff

I am a student in a Cyber Security program, at my local community college, and a long-time Linux user. Tonight in class, we were working to install *Snort* into a Fedora build, and another student suggested we use *Security Onion*.

After some thought and research at home, I began to wonder if *Linux Format* had ever written up either *Snort* or *Security Onion*, and it appears that the answer is no.

That lead me to wonder if you had ever thought about doing reviews on the

various tools used to ensure security, like *Snort* or *Security Onion*.

I know you have distributed Kali; how about an issue focused on the tools used by ethical hackers? I realise the line is pretty narrow between the ethical and non-ethical hacker; but you can't control what anyone does with the knowledge contained in your magazine. Thanks for the great magazine and your time.

Jonathan Heard, Phoenix, AZ

Neil says: Thanks for the suggestions. There's an almost endless supply of security tools out there, so it's genuinely

useful to hear what readers have had success with and what they haven't. We'll pencil in some time to have a look at these at some point down the line!



› We're pretty sure we'd remember *Snort* if we'd come across its logo before...



Write to us

Do you have a burning Linux-related issue you want to discuss? Want to tell us what we've done wrong again or what we should be looking at next? Or just want to suggest future content? Then write to us at *Linux Format*, Future Publishing, Quay House, The Ambury, Bath, BA1 1UA or lxflatters@futurenet.com.

T3

The Gadget Magazine

MORE
GREAT
CONTENT
AT **T3.COM**

GET FIT FAST IN 2016
WITH THE VERY BEST TECH
FOR RUNNING,
CYCLING AND MORE...



LIFE'S BETTER WITH T3

www.myfavouritemagazines.co.uk/t3





United Linux!

The intrepid **Les Pounder** brings you the latest community and LUG news.

Find and join a LUG

» Alpinux, le LUG de Savoie

Meet on the 1st and 3rd Thursday of the month at the Maison des Associations de Chambéry.

www.alpinux.org

» Bristol Hackspace

Studio G11, 37 Philip Street, Bedminster, Bristol, UK, BS3 4EA

<http://bristol.hackspace.org.uk>

» Cheltenham Hackspace

The Runnings Trading Estate, Cheltenham. Thursday from 7pm.

www.cheltenhamhackspace.org

» Huddersfield Raspberry Jam

Meet 4th Saturday of the month at Huddersfield Library.

<https://huddersfieldraspberrypjam.co.uk>

» Hull Raspberry Jam

Malet Lambert School, Hull. Every other month. See Twitter for details.

<https://twitter.com/hullraspbjam>

» Lancaster and Morecambe Makers

Unit 5, Sharpes Mill, White Cross, Lancaster, Wednesday evenings 6.30pm till late.

<https://lamm.space>

» North Kent Raspberry Pi User Group

Every two weeks at Medway Makers, 12 Dunlin Drive, St Mary's Island, Chatham ME4 3JE

<https://nkrpug.wordpress.com>

» Preston Hackspace

28A Good St, PR2 8UX.

Open night is 2nd Monday of the month from 7pm.

<http://prestonhackspace.org.uk>

» Surrey and Hampshire Makerspace

Tuesday and Friday evenings, Boileroom in Guildford.

www.shmakerspace.org

The Preston Raspberry Jam

Where the kids are making the waves.

When we first went to Preston Raspberry Jam, it was the very first event way back in July 2012, mere months after the Raspberry Pi had first been released. Organised by Alan O'Donohoe, then a local teacher and passionate Computing advocate. Alan created the Raspberry Jam concept as a means to enable access to all, as at that time Raspberry Pi were exceptionally hard to come by. At this first event, the demographic was those that had grown up with computers in the 1970–1980s.

Alan has since moved away from teaching and now works to help teachers across the UK to learn computing skills, but he has remained the figurehead for the Jam. We popped along to the latest Jam and noticed that the demographic has shifted: Children are everywhere sat at Pis learning new skills in Python and Scratch and tables of adults are showing off their new projects to eager children.

It was lovely to see that two junior members of the

community, Joshua and Elise were there and presenting their latest projects to an eager audience, these children are not even teenagers yet and already they are making waves in the Pi Community.

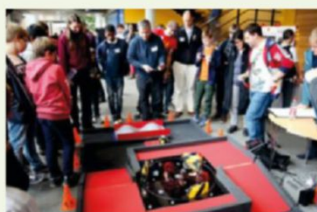
The Raspberry Jam concept is essential to the success of the Raspberry Pi. Without Jams, like the Preston one promoting the benefits of computing, working together and helping each other, the Pi would be just another single board computer.

Preston Raspberry Jam meets on the first Monday of each month, and they have an Eventbrite page where you can sign up: <http://bit.ly/PrestonRaspberryJamFeb17>. **LXF**



» Preston raspberry Jam is where young and old hackers meet to pass on knowledge.

Community events news



SCaLE 15x

The Southern California Linux Expo, SCaLE for short, is a multi-day Linux conference that seeks to educate and inspire those that are looking to work with open source software. Whether you're

an old hand with Linux or a complete newcomer, this event will have something to interest you from talks hosted by *Linux Format* alumnus Jono "Mr Community" Bacon or from influential speakers such as Brendan Gregg Performance Architect for Netflix.

Taking place on March 2-5 at the Pasadena Convention Center this event has become a staple of the conference calendar.

<http://www.socallinuxexpo.org/scale/15x>

Pi Wars

The Raspberry Pi community has also created their own less violent version of *Robot Wars* for the Pi. The idea developed by the team behind Cambridge Raspberry Jam, sees competitors from around the UK take part in various trials and battles. Robots are built to strict size and weight requirements. The next Pi Wars is on April 1-2 2017 in Cambridge and tickets are on now sale.

<http://piwars.org>

FLOSS UK Spring 2017

FLOSS UK has long helped to educate, inform and push forward the knowledge of system and network admins. This three-day conference in Manchester takes place from March 14-16 with tickets costing around £150. As well as conference talks from speakers in the community, there are a series of all day personal development workshops. You can learn more, and purchase tickets via their website.

<http://bit.ly/FlossSpring2017>



The home of technology

techradar.com

LINUX FORMAT Reviews

All the latest software and hardware reviewed and rated by our experts

Intel Core i7-7700K

How good are the eighth-generation waters?
Zak Storey dips his toe in Kaby Lake to find out.

In brief...

- ▶ **Socket:** 1151
- ▶ **Cache:** 8MB
- ▶ **Type:** 64-bit
- ▶ **SSE:** 4.1/4.2, AVX 2.0
- ▶ **Process:** 14nm
- ▶ **Cores:** 4
- ▶ **Threads:** 8
- ▶ **Clock:** 4.2GHz
- ▶ **Turbo:** 4.5GHz
- ▶ **TDP:** 91W
- ▶ **Max memory:** 64GB DDR4
- ▶ **Channels:** 2
- ▶ **GPU:** Intel HD Graphics 630
- ▶ **Clock:** 350MHz
- ▶ **Max:** 1.15GHz
- ▶ **Units:** 24
- ▶ **OpenGL:** 4.4
- ▶ **Displays:** 3
- ▶ **Virtual tech:** VT-x, VT-d

Kaby Lake is the eighth processor iteration since the introduction of the Intel Core series that started with Nehalem. For eight years, the company has pressed to push the advantage in its processor lineup, and each and every time it's managed a marginal 10–15 per cent performance increase. This process has been based around the concept of Tick-Tock. A new architecture would be designed based on the latest transistor size, then that transistor size would be shrunk the following year. For instance, Sandy Bridge (or the Core i5-2500K) held the new architecture, while Ivy Bridge (Core i5-3570K), released a year later, was the die shrink, and so on.

However, time and time again, Intel has come up against issues. The first we saw of this was with the Haswell refresh, known as Devil's Canyon, then once more as Broadwell was delayed for six months – each drop in transistor size becoming ever more difficult to achieve. Fast-forward to the release of Skylake, Intel's first 14nm architecture, and we're greeted with news that Tick-Tock is being annexed in favour of a new scheme called PAO, or Process, Architecture, Optimisation. In short, the die shrink (the Tick) turned into the Process; the architecture (the Tock) is now, well, the Architecture; and lastly we have Optimisation. A new piece to the puzzle, where Intel tries to gain the maximum performance possible from a mature manufacturing process and a more optimised architecture. On top of giving Intel an extra year to perfect its processes, it also gives us another chip.

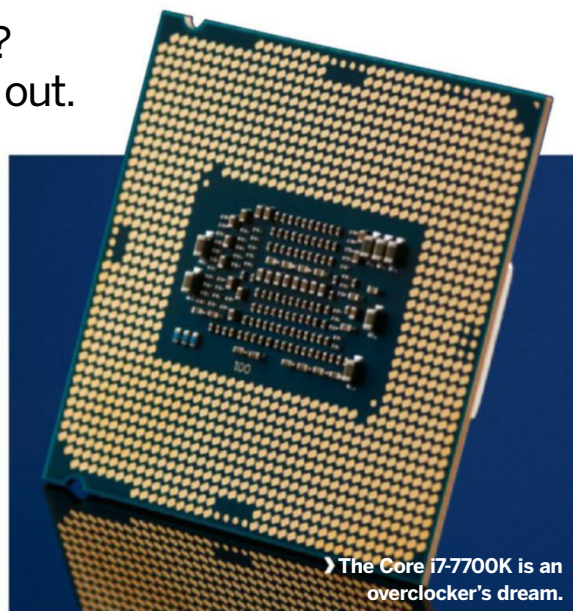
Ignoring Devil's Canyon, Kaby Lake is the first true Optimisation release, and with it comes a lot of questions. If Intel's generational gains have been so minimal, what can Kaby Lake do to

make that any different? Well, that's what we're here to answer.

Intel's Core i7-7700K is a four-core, eight-threaded, low-power rendering powerhouse. The pinnacle of what Intel has managed to achieve with Skylake and the 14nm technology. With greater performance and better overclocking potential than we've seen from any of Intel's last few generations, it comes packing a whopping 4.2GHz core frequency, turboing up to 4.5GHz with boost. We were immediately impressed with its out-of-box performance. In Cinebench R15 (via Wine), we saw scores well into the high 900s, with single-core performance peaking at 194 – a sweet 8 per cent increase over Skylake. It was a similar experience across the board.

What really impressed, though, was its overclocking potential. We increased the multiplier to 48 without the core batting an eyelid, and stock voltages happily keeping the 4.8GHz chip on track. 5GHz came next, needing only a 0.05V increase to the Vcore, with temperatures sitting comfortably at 62° C under our 280mm NZXT Kraken X61. But it kept going, higher and higher, until we topped out at 5.2GHz with 1.4V added to the Vcore – a substantial increase, but temperatures still only sitting at 80° C.

Is it worth upgrading today? Well, that depends on what interests you. Generally speaking, the Z270 chipset is



▶ The Core i7-7700K is an overclocker's dream.

feature-rich, and adds additional support for PCIe devices and such. But it pales in comparison to the change from Z97 to Z170. Putting them side by side, the difference between Skylake and Kaby Lake is minimal. If you're already set up with the sixth generation of processor, it's not worth your time; upgrading from Ivy Bridge, Haswell, or Devil's Canyon, on the other hand, very much is. And we can't recommend this core enough in that regard. **LXF**

LINUX FORMAT Verdict

Intel Core i7-7700K

Developer: Intel
Web: www.intel.com
Price: £340

| | |
|-------------|------|
| Features | 9/10 |
| Performance | 8/10 |
| Ease of use | 9/10 |
| Value | 7/10 |

▶ You might want to hold off until our Ryzen review, but this is a solid buy with excellent overclocking abilities.

Rating 8/10

Asus GTX 1050 Ti

Not a Star Wars fan, **Zak Storey** warns you that this isn't the budget graphics card you're looking for.

In brief...

- » GPU: Pascal (GP107-400)
- » Process: 14nm
- » Transistors: 3.3 billion
- » CUDA cores: 768
- » Core: 1,290MHz
- » Memory: 4GB GDDR5
- » Bus: 128-bit
- » TDP: 75W
- » Ports: 1x HDMI 2.0b, 1x DisplayPort 1.4 (ready), 2x DVI
- » Power: 1x 6-pin

Nvidia's transition across to the 16nm FinFET manufacturing process brought with it huge leaps for the vast majority of its GPUs. The power-hungry GTX 1080 showed us that gaming at 1440p with high refresh rates was possible off a single card; the GTX 1070 packed the power of a Titan into an affordable price point; and the GTX 1060 provided as much performance as last tier's first flagship at a lower cost than a good meal out.

It was all going so well: clock speeds skyrocketed past 2GHz, CUDA core counts shot up... Alas, it seems the mighty green giant had to stumble somewhere, and that somewhere is with the Asus ROG Strix GTX 1050 Ti, to give it its full name.

Traditionally, the X50 series has been an awe-inspiring thing of beauty. Whether it's the 750 Ti's powerless design, or the GTX 950's fantastic price to performance ratio, Nvidia really pushed those boundaries, proving time and time again that you could innovate in the mid-to-low range just as much as the high end. So what's so bad about the GTX 1050 Ti that it warrants such a scalding paragraph? Let's find out.

The GPU powering the heart of the GTX 1050 Ti comes packing a whopping 768 CUDA cores, 48 texture units and 32 ROPs. Compare that to the GTX 950, and you'll see that it also had 768 CUDA cores, 48 texture units and 32 ROPs. But then, that's fair, right? It's a process shrink; it's not something you'd expect to change. We



» A solid card, but there are better alternatives.

would be inclined to agree with you if it weren't for how Nvidia has specced the rest of its 16nm lineup. The GTX 1060, for instance, increased its CUDA core count from 1,024 to 1,280, forcing the transistor count to increase by 1.46 billion in the process.

Take the card out of the ecosystem entirely, however, and you're greeted with a powerful GPU more than capable of tackling the latest games at 1080p Ultra settings comfortably. The problem arises when you look just slightly up the chain at the GTX 1060 3GB. The fact is, you can currently buy a compact GTX 1060 3GB for exactly the same price (£190/\$240) as this card. And if you did, your frame rates would increase by 12–19 frames per second on average at 1080p. That's a huge difference, and it's something that really puts us off recommending this card.

We know what you're thinking, though. The GTX 1050 Ti has 4GB of memory compared to the 1060's 3GB. Surely that counts for something here? Well, to be honest, not that much. When testing our titles, we noticed frame rates across all three major resolutions varied by 1–3 frames per second when it came to our 3GB and 6GB variants of the 1060, and that's something we can mostly dial down to the 3GB variant's fewer CUDA cores.

The Asus ROG Strix GTX 1050 Ti's aftermarket solution is a crisp reminder that you can innovate and

design a graphics card that still looks classy at a respectable price point. The dual-fan cooler is subtle and quiet, with 0dB fan technology ensuring temperatures remain low and noise is minimal, and the included backplate and hint of RGB AURA tech that Asus is so well known for glams up an otherwise budget purchase, making this a truly solid aftermarket card.

If you've not upgraded for some time (we're talking pre-GTX 600 series), the GTX 1050 Ti is a fantastic solution to alleviate all of your 1080p gaming woes. However, if you've got the cash, you should spring for the 3GB GTX 1060 variant instead. For the time being, 3GB is more than plenty for 1080p. Anything more than that and you're going to need far more processing power anyway. **LXF**



» The 0dB dual-fan setup keeps things cool and quiet.

LINUX Verdict

Asus ROG Strix GTX 1050 Ti

Developer: Nvidia
Web: www.nvidia.com
Price: £190

| | |
|-------------|------|
| Features | 8/10 |
| Performance | 6/10 |
| Ease of use | 9/10 |
| Value | 5/10 |

» The GTX 1050 Ti fails to innovate, and you'll feel the brunt of that decision with the 1060 models at a similar price.

Rating 6/10

Synology DS916+

A new generation of NAS boxes is pushing the envelope of what it means to be attached storage. **Neil Mohr** is impressed.

Specs

- » **CPU:** Intel Pentium N3710 quad-core 1.6GHz
- » **Memory:** 2GB or 8GB
- » **Network:** 2x Gigabit
- » **Bays:** 4x SATA III (hot swappable)
- » **FS:** Btrfs, Ext 4
- » **RAID:** 0, 1, 5, 6, 10, JBOD
- » **iSCSI:** 32 targets
- » **Virtual support:** VMware 5/6, Hyper-V, Citrix, OpenStack
- » **IP cam:** 40 (+ licences)
- » **Ports:** 1x eSATA, 3x USB 3.0
- » **Video:** H.264 (AVC), H.265 (HEVC), MPEG-2 and VC-1
- » **Size:** 165 x 203 x 233.2mm, 2.04kg

Synology is renowned for its network attached storage (NAS) solutions, and this latest DS916+ model is aimed at professionals and smaller businesses that are looking for a highly adept four-bay storage solution, with power, security and flexibility at its heart.

The Intel Pentium N3710 1.6GHz quad-core processor it's based on should give you an indication of the ambitions Synology has for this model. It's a quad-core, 64-bit processor, with VT-x virtualisation, up to 8GB of memory support and QuickSync video encoding. You're not going to put a powerful processor into a NAS unless you plan to use it.

It offers four hot-swap SATA III bays, with a maximum capacity of 40TB, an eSATA port enables expansion, with a storage pod increasing to a maximum 90TB capacity. Internally, the system uses Btrfs or Ext4, while external drive supports extend to Ext3, FAT, NTFS, HFS+ and exFAT (a paid-for extra).

Supporting its higher-end credibility are the twin Gigabit Ethernet ports, with Link Aggregation and Fail-over support. Full access power usage peaks at 30W and drops to 13W once the drives spin down.

Setup is a snap, either via a 'search' URL, directly on port 5000, or via the *Linux Synology Assistant* tool with Deb and RPM packages. Synology provides all its tools packaged for Ubuntu and Fedora, alongside Mac and Windows, which is nice. Using Btrfs by default,



Synology Hybrid RAID is the standard option, but JBOD, RAID 0 to 6, plus 10 are all available as options. It also supports full snapshot capabilities, something that comes into its own when you start exploiting its virtualisation capabilities.

A key element to the DSM 6.0 OS is its add-in package system. It enables you to add Synology and third-party features, from backup solutions, video security software, antivirus and *Plex* media streamer to *CalDAV* web calendar services. There's around 50 official packages and about the same number of third-party ones.

Full backup support has always been a key feature and still is. The new *Hyper Backup* tool offers a single point to manage all your backup and recovery tasks – local, remote, *Rsync* and a host of cloud options are here, with full accelerated encryption. Cloud sync and *Cloud Station* provide further abilities to synchronise data between local and cloud-based accounts.

Managing the device is a cinch, thanks to the ever-improving *Synology DiskStation Manager 6*. It's a web-based graphical interface, effectively a full desktop in your browser. Making it easy to manage, adjust, add and remove services. There's the option to SSH in, too.

Making the latest build even more flexible is virtualisation support, via its own *Virtual DSM Manager* and, more usefully, a Docker extension. The former enables you to run more than one instance of the Synology DSM, the

» **Worth every penny of that high price.**

latter Docker option brings with it a gaggle of pre-built Docker containers, from Ubuntu and CentOS to Ghost and MariaDB, direct from the Docker repository. Just add the one you want, specify resources and stand it up, ready to access.

With the 8GB model costing just £18 or so more than the 2GB one, it's this type of exceptional feature that helps the DS916+ stand out from the crowd. You might balk somewhat at the seemingly high price (for which you could get an entire new desktop) but the level of features, support and ease of use provided by the Synology is why professionals are willing to pay the price. A superb pro-level NAS that's hard to fault in any way. **LXF**



Features at a glance



DSM OS 6.x

The DSM interface is attractive, simple to use, easy to expand and feature-packed.



Pentium N3710

A key part of the performance is the use of the capable quad-core Pentium processor.

LINUX Verdict

DiskStation DS916+

Developer: Synology
Web: www.synology.co.uk
Price: £590 8GB (£492 2GB)

| | |
|--------------------|-------|
| Features | 10/10 |
| Performance | 10/10 |
| Ease of use | 7/10 |
| Value | 8/10 |

» *An accomplished NAS that's easy to set up and administer, and has boundless features and performance.*

Rating 9/10

TW: Warhammer

You might think you know rock, paper, scissors, but **Jody Macgregor** says this is a game where paper can beat scissors, so long as there's enough of it.

Specs

Minimum:

» **OS:** Ubuntu 16.10 64-bit, SteamOS 2.0
 » **CPU:** 3.4GHz Intel Core i3-4130 or 3.5GHz AMD FX6300

» **Memory:** 4GB

» **HDD:** 29GB

» **GPU:** 1GB Nvidia 650Ti (driver 367.28), AMD R9 270 (Mesa 13.0.1), 2GB VRAM

Recommended:

» **CPU:** 3.4GHz Intel Core i7-4770

» **Memory:** 8GB

» **HDD:** 35GB

» **GPU:** 4GB Nvidia 970 (driver 367.28), AMD RX 480 (Mesa 13.0.1), 4GB VRAM

The *Warhammer* world is a fantasy setting, loosely based on Renaissance Europe, but with the fiction of JRR Tolkien funnelled into it, while copies of *2000 AD* covers are scattered on top. It's a mish-mash of everything someone at Games Workshop ever thought was cool, and it's both familiar and really weird.

This is a ground-up alteration of the *Total War* formula, to make it suit the fantasy setting of the Old World. That formula comes in two halves. The first is a turn-based grand campaign about marching armies across a map, and managing provinces through construction, research and taxation. The second happens when those armies meet and drop into a real-time – though pausable – battle.

In previous *Total War* games, the factions played in a relatively similar way, but not any more. For instance, the Greenskins have a meter measuring each army's Fightiness. Win battles, and it rises. Lose or squat in your hovels like a coward, and it drops. Dwarfs, on the other hand, have to keep track of grudges. They never forget a slight, carefully noting each in a massive book of bitterness.

All these differences dramatically affect the way you play. As the Greenskins, you fight just to keep armies Fighty, and raid neighbours without regard. It's not the kind of game where replaying as a different side means 'focusing slightly more on missile weapons'. Each faction is almost a different game.

There's variance between factions in both the turn-based campaign game and the real-time battles. Dwarfs don't have wizards but are blessed with



» Real history is given the brush-off in favour of glorious fantasy.

plenty of artillery. Vampire Counts don't have missile units – for them, it's all about lurching forward, targeting specific enemies with flying units and Black Knights, while the skeletons and zombies shamble up to fill the gaps. The Greenskins have a bit of everything, but can be hard to control.

It's possible to pull off amazing things even in the morass, though it helps to abuse the slow-motion button and give orders while paused, as you can in single-player mode. You can overcome odds that the auto-resolve option for battles isn't able to.

At the start of a campaign, you choose which of two Legendary Lords will lead your faction – iconic *Warhammer* characters like Emperor Karl Franz and High Wizard Balthasar Gelt – with the other becoming available during the campaign.

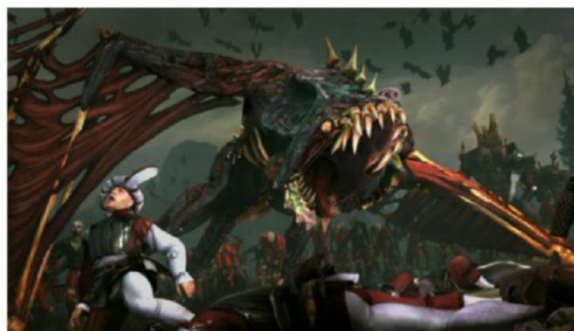
Because quests are bespoke little stories separate from the campaign – you use your regular army but opponents are conjured up on the spot, rather than drawn from existing enemies – there's a risk of them seeming inconsequential. But their unique nature, and the few paragraphs of narrative that come with quests, are reward enough that you may waste too much effort chasing them.

There are unique tech trees to research, buildings to construct, public order to maintain, and diplomacy to tinker with. Even the Greenskin tribes

engage in limited diplomacy – though they don't make trade agreements, they do negotiate alliances with each other, and sometimes the Dwarfs tempt them with gold to buy peace.

One of the most *Warhammer*-ish things about it is Chaos. After 20 turns, warnings appear: Chaos gathers in The North. It's another 50 turns before we notice their effects, a spreading corruption like the undead's. It's past turn 100 before we engage with them, but by then The Northern Old World is in ruins, and Archagon The Everchosen leads a doomstack right towards us.

The best *Total War* games are the most focused, whether on a single nation or a single general. *Warhammer* takes in a continent but tells one story, and it's potent because of that. **LXF**



LINUX FORMAT Verdict

Total War: Warhammer

Developer: Feral Interactive
Web: www.feralinteractive.com
Licence: £39.99

| | |
|------------------|------|
| Gameplay | 9/10 |
| Graphics | 8/10 |
| Longevity | 7/10 |
| Value | 6/10 |

» If you find history bland compared to nonsense made up by strange Brits, *Warhammer* is the *Total War* for you.

Rating 8/10

lifehacker | UK

Helping you live better & work smarter



LIFEHACKER UK IS THE EXPERT GUIDE FOR ANYONE LOOKING TO GET THINGS DONE

- Thousands of tips to improve your home & workplace
- Get more from your smartphone, tablet & computer
- Be more efficient and increase your productivity

www.lifehacker.co.uk

twitter.com/lifehackeruk facebook.com/lifehackeruk

Roundup

» Every month we compare tons of stuff so you don't have to!

Desktop BSDs

Looking to boost his geek cred, **Mayank Sharma** wonders if he can replace his desktop Linux distribution with a BSD-based one? [Don't be silly—Ed]



How we tested...

A BSD OS isn't the same as a Linux distribution (distro) which is to say that you do things differently under BSDs as compared to what you are used to under Linux. This is why the primary goal of the Roundup is to find a BSD that involves the gentlest learning curve. Using an OS starts with the install and we'll evaluate the various BSD installers to find the one that's simple enough to navigate for first timers, while offering options to satisfy experienced campaigners.

Additionally, because of the differences between Linux and BSD, we'll be looking for documentation that helps bridge the gap. We'll also up scores if an OS supplies easy to use admin tools. Finally, access to everyday open source applications, both within the OS and in the software repositories (repos) is another crucial factor that'll have a bearing on the final result.

Our selection

- » DragonFly BSD
- » GhostBSD
- » MidnightBSD
- » NetBSD
- » TrueOS

For the distro hoppers among us, a Linux distro is just a collection of applications and utilities. We can be productive with any distro as long as it gives us access to our cherished and trusted tools. So how about a diversion into the land of the BSDs? While they haven't caught the fancy of the mainstream tech press, BSDs are known for their robustness, reliability and security and are fairly popular with sysadmins. That said, you can slap popular open source applications on top and use BSDs for everyday desktop computing tasks, such as browsing the web, listening to

music, watching DVDs, playing games and reading PDFs.

The contemporary BSDs that we have on test can be traced back to the 1970s. BSD stands for the Berkeley Software Distribution and it was the name of the toolkit of enhancements for Unix that was created at the University of California, Berkeley. In contrast to Unix, which was developed

at Bell Labs, BSD was created by students and university faculty. BSD was distributed as a package of software enhancements for Unix and over time became a usable OS in itself. The current stable of BSD distros are a family of OSes that are derived from the original and we'll evaluate the strengths of the most popular desktop BSDs and help you pick one that's easy to use.

“We'll evaluate the strengths of the most popular desktop BSDs and help you pick one.”

Core strengths

My BSD is better than yours.

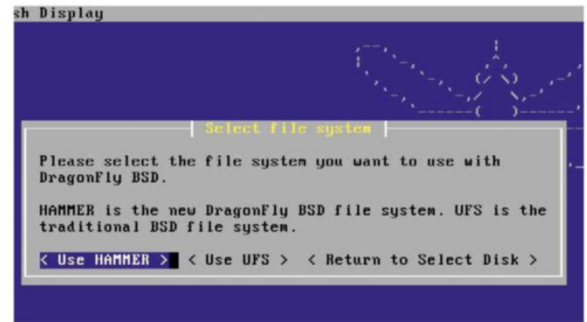
Before you earmark a machine for BSD, you should know the strengths of the candidates that are on offer and what makes them suitable as a desktop operating system. DragonFly BSD is a popular fork of FreeBSD that's now developed in a direction of its own and is considered one of the main BSD distros. The OS has diverged significantly from FreeBSD and is popular for its implementation of virtual kernels and a feature-rich 64-bit filesystem called HAMMER, which has built-in mirroring, instant crash recovery and historic access functionality. It's also popular for its Sun ZFS-like features but with a friendlier licence.

MidnightBSD also owes its origin to FreeBSD and in its decade-old existence has imported features from DragonFly BSD, OpenBSD and NetBSD. Its goal is to create a system that appeals to both beginners as well as more experienced BSD users. The OS

targets 32-bit and 64-bit computers and supports all hardware supported in FreeBSD 9.1. MidnightBSD has also created its own custom package management system known as *mports*.

NetBSD is without a doubt the most portable BSD that you can use on virtually all kinds of hardware: from a toaster to the International Space Station (ISS). NetBSD's rock-solid foundation and portability were a big draw for NASA, which used it for a project on the ISS. Thanks to its extensive hardware support there's also a good chance you can run a NetBSD-based desktop on the oldest computer in your attic.

GhostBSD and TrueOS are often pitted against each other because of the pair's similarities: They are both BSDs derived from FreeBSD and provide a graphical desktop straight out-of-the-box. GhostBSD is built with C, Python, *GTK* and *Bourne Shell* (sh) while TrueOS is built with C++ and *Qt*.



» **DragonFly uses a home-brewed HAMMER filesystem that features instant crash recovery and historical snapshots.**

However one key difference is that GhostBSD is distributed as both 32-bit and 64-bit while TrueOS is 64-bit only. GhostBSD also has minimal requirements compared to TrueOS which needs at least 4GB of RAM since it uses ZFS. To offset this hardware requirement, TrueOS users benefit from its support for newer Intel graphics chipsets and hardware. It also outscores its peers with unique features such as *PersonaCrypt*, which allows for the encryption of a user's **home** directory and carrying it to other TrueOS machines along with GELI full disk encryption.

Verdict

DragonFly BSD

★★

GhostBSD

★★★★

MidnightBSD

★★

NetBSD

★★★

TrueOS

★★★★

» *GhostBSD and TrueOS stand out as making the desktop their primary focus.*

Out of the box experience

Do they package enough to get you going?

Since we are planning to use these OSes on the desktop, the first impression is paramount and we have to evaluate them on the basis of what they offer right after you boot each OS for the first time. DragonFly BSD, MidnightBSD and NetBSD all offer a similar initial user experience. Their respective live CDs

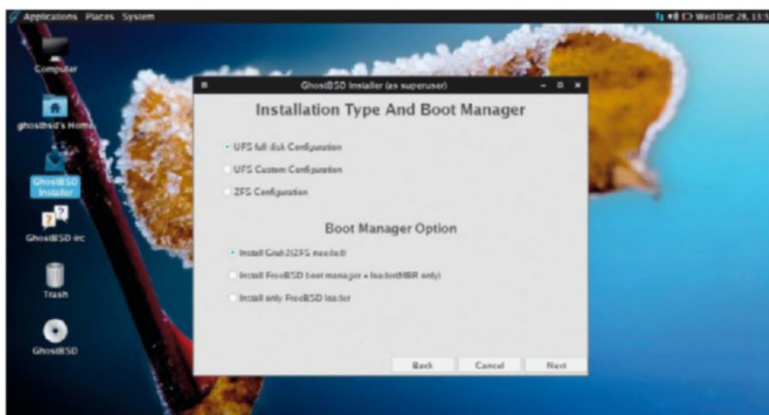
boot to a console which might come as a shock to first timers. Not only do you not get a graphical desktop, you'll also have to navigate through text-based OS installers.

You still don't get a graphical desktop environment with DragonFly BSD and have to labour through the documentation to install one. Despite

all this trouble, the OS still pitches itself as a desktop, albeit one that you'll have to build from scratch yourself. On the upside however, unlike Linux Arch, crafting a DragonFly BSD desktop is rather simple and straightforward thanks to the availability of binaries.

The other two OSes also offer a similar experience with their own peculiarities. For example, in NetBSD you'll have to install the binary package manage yourself before pulling together the components for a graphical desktop.

TrueOS also boots straight into the installer which then helps you anchor a fully functional OS with the Lumina desktop environment. Sure, it looks different but it is at least dotted with familiar applications that ensure full productivity. The default GhostBSD experience is the closest to what a Linux user would expect; the live CD boots into a graphical desktop, which can be either MATE or Xfce depending on the image you've downloaded.



» **GhostBSD ships all the regular desktop applications including Firefox, LibreOffice, Shotwell, Thunderbird and GNOME MPlayer.**

Verdict

DragonFly BSD

★★

GhostBSD

★★★★

MidnightBSD

★★

NetBSD

★★★

TrueOS

★★★★

» *GhostBSD is the only one that boots a live DC into a graphical desktop at start.*

Linux-like experience

Déjà vu?

We've been harping on about the fact that BSD isn't Linux. But having said that, are these BSD OSes really all that different from your regular Linux distribution? Surely on the inside, but on the outside they run popular open source

desktops, such as KDE and Gnome and use regular mainstream applications. But there's more to an operating system than the visual presentation. Despite the familiarity of applications and the comfort of desktop environments, running and managing BSD

OSes isn't the same as your typical Linux distribution. In this section, we're on the lookout for the BSD OS that goes the extra mile, particular for new users, and makes an effort for you to be able to use them for everyday computing tasks.

DragonFly BSD ★★★★★

The installation medium for DragonFly is available for 64-bit architectures only. The project releases an ISO image for optical drives and an IMG file for installations via USB. Both boot into a live environment and enable you to log in as root and check the compatibility of your hardware from the command line. Once you're satisfied you can fire up the installer. DragonFly doesn't install a graphical desktop, but the latest version of the OS supports GPUs from the Haswell family and OpenGL acceleration is available out of the box on supported i915 and Radeon GPUs. To set up a desktop you'll have to follow straightforward instructions on the project's website. It doesn't take much to have a fully functional desktop. However, while DragonFly does focus on desktop users, it requires considerable experience on the command line.



GhostBSD ★★★★★

Here's a BSD that closely resembles a desktop Linux distro in form and function. Unlike most of the other BSDs on test here, GhostBSD boots straight into a live graphical desktop environment. Earlier versions of the OS shipped multiple desktops, but the current release is available in MATE and Xfce-based flavours. The multilingual graphical installer offers an automated partitioning scheme if you wish GhostBSD to take over the entire disk and eases the process of getting the OS onto the hard drive. The default installation includes all the applications you need to go about your daily desktop computing tasks. You can also easily flesh it out using the graphical package manager which is really convenient. Despite all the preinstalled apps, GhostBSD is still quite quick off the mark. If you look past its package manager, you wouldn't even realise that you're not running a MATE/Xfce-based Linux distro.

Documentation & support

There is no escaping having to RTFM.

Good documentation and support go a long way in helping adopt any BSD OSes. MidnightBSD doesn't seem to agree with this notion; the project has an official guide that helps install the OS and work with its *mports* package management system, but for everything else it directs users to the FreeBSD handbook. Alternatively, you can look for help on the mailing list or direct any queries to the developers at the published email address.

GhostBSD is marginally better. Its documentation is in a wiki, but there's not much of it besides a FAQ and a handbook that's still being written. In addition to the IRC channel, the project has fairly active forums to help sort issues and share tips and tricks.

The majority of the documentation for NetBSD is FAQs. There are various architecture-specific FAQs as well as a guide to help you install and admin the OS. The website also hosts documentation on topics such as

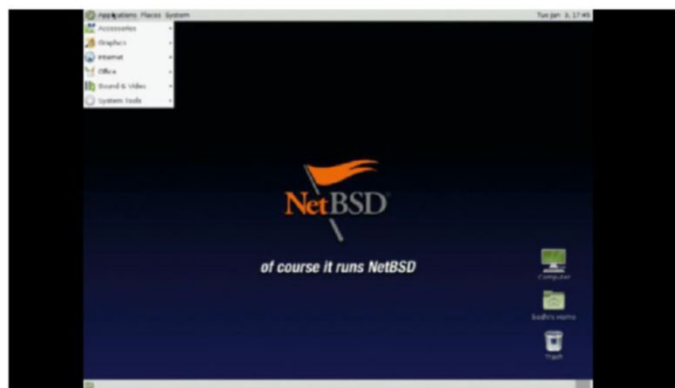
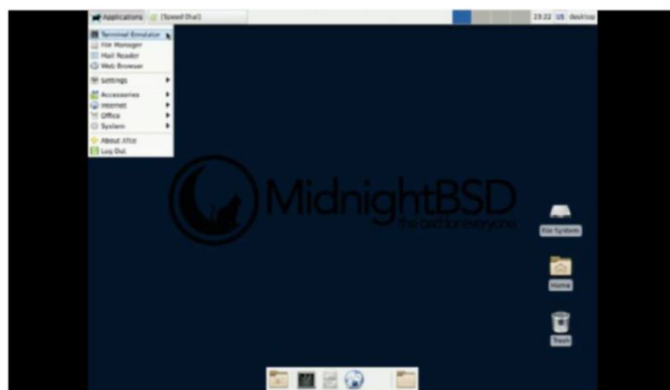
networking and virtualisation. For extra help, you can use the NetBSD-specific boards on daemonforums.org or engage with the developers either via IRC or mailing lists. In contrast, TrueOS has a comprehensive handbook that covers advanced installation topics along with a chapter on the Lumina desktop. For support, TrueOS uses Gitter, an IRC-like instant messaging service and its own subreddit. Similarly, the DragonFly website's documentation section is a wonderful starting point.

Verdict

DragonFly BSD ★★★★★
GhostBSD ★★★★★
MidnightBSD ★★★★★
NetBSD ★★★★★
TrueOS ★★★★★
» DragonFly makes up for its steep learning curve with great documentation.

MidnightBSD ★★☆☆☆

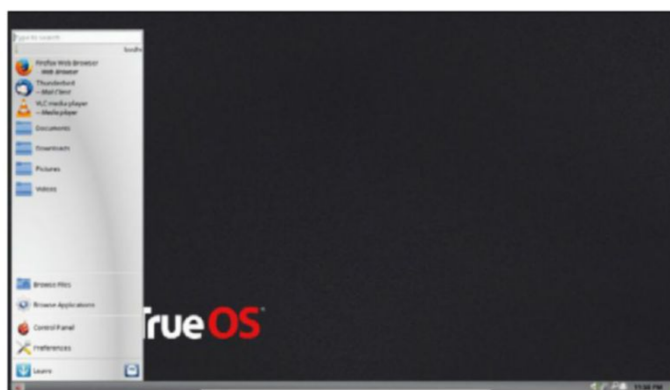
Like some of its peers, if you decide to use MidnightBSD don't expect to be treated with kid gloves. Despite its focus on the desktop users, be prepared to labour through the documentation to build your desktop. Its *ports* collection is also fairly limited. The OS is slowly progressing to its goal and while it's been under development for over 10 years, the low version number (0.8) gives away its developmental status. You can follow the instructions in its manual to add to the fairly minimal default installation. However, in our testing, MidnightBSD was the only one that consistently threw errors while installing packages which meant we couldn't test its new experimental graphical package manager. It's a different story once it's all setup, the bootup speed and application launches are lightning quick even compared to other BSDs.

**NetBSD** ★★☆☆☆

The OS is esoteric from the get-go. Like several others, it uses a command-line installer that's too verbose. You'll also have to get used to BSD's nomenclature for device names and partitions if you plan to get anywhere with it—editing files under */etc* to get the wireless card to connect brought back some rather bittersweet memories. You'll have to add users from the command line and create mount points and mount optical discs manually. Another thing you'll have to get familiar with is NetBSD's use of the *rc.d* system to control services, which is similar to System V but without run levels. However, once you pull-in the graphical desktop things seem pretty normal as you have access to several popular open source packages.

TrueOS ★★★★★

FreeBSD is one of the most widely recognised BSD alternatives and its popularity has spawned several derivatives. PC-BSD, which is now called TrueOS, is one such descendant that's made a name for itself by extending FreeBSD's famed stability to everyday desktop users. TrueOS employs everything that's familiar to a Linux desktop user including a graphical installer, a graphical desktop environment and graphical admin tools. The installer offers good defaults for new users as well as plenty of options for advanced users. By default, the OS installs the Lumina desktop which is fairly intuitive to navigate. Post-install, TrueOS takes you through a series of steps to set up your computer. The OS also includes several custom tools to ease management including useful ones such as encryption.



Back to school

Do using these OSes force you to relearn?

Here we are switching to a completely different OS rather than trying out a different Linux distro; the differences are going to be quite considerable. Again, DragonFly BSD, MidnightBSD and NetBSD are as discrete as they come. Everything about these OSes: from their console-centric live environment to the instructions in their text-based installers to the partition names, the filesystem and their package managers all require some getting used to.

You can't use any of these BSDs without first reading the documentation. Even after you're done with them, you need to be prepared to visit forums to troubleshoot as you ease your way to the desktop on either of these OSes. Yet despite all of this, we didn't have one error-free install on MidnightBSD, either on virtual or physical hardware.

You'll have considerably less difficulty with the other two BSDs. Thanks to the Lumina desktop, TrueOS makes you conscious that you are using

a system that's not Linux. However, you can easily take the time to climb the slight learning curve thanks to the availability of familiar applications. With GhostBSD, you virtually don't notice any difference, besides the obvious ones between BSD and Linux. In fact, you can even ask GhostBSD's installer to install the *GRUB2* boot loader instead of FreeBSD boot manager for more familiarity. Similarly, you can choose to use the *Bash* shell instead of the default *fish* shell.

Verdict**DragonFly BSD**

★

GhostBSD

★★★★★

MidnightBSD

★

NetBSD

★

TrueOS

★★★★★

» Stick to GhostBSD for a Linux-like desktop experience.

Ease of installation

What does it take to get them up and running?

When you do finally decide to try a BSD-based desktop, you'll be well advised to test the OS inside the comforts of a virtual machine before subjecting it to a physical disk.

DragonFly BSD has a menu-driven text-based installer. Like the others, you can install it without much effort on a

machine where it's the sole OS on the hard disk. In fact, we'd advise you to not install any of these OSes on a multi-boot computer without first getting comfortable with their jargon. You can use the default options to successfully install any of these BSDs onto the disk. However, they all offer the flexibility to enable you to make informed choices

especially if you're familiar with the BSD filesystems.

Similarly, installing MidnightBSD is a laborious task unless you are planning to let it take over the entire disk. The OS calls on a post-installation script to earmark services to start at boot and to create users.

NetBSD uses an *ncurses*-based, menu-driven installer that's fairly verbose compared to the others. Just like the others, the partitioning steps are particularly cumbersome to navigate. At the end of the install, you'll be asked to configure some essential aspects of the installation. Also, while the full installation scheme installs the basic X window system components, it doesn't include a graphical desktop.

Once again, TrueOS and GhostBSD stand apart in that they both employ graphical installers. Like their peers, both offer automated partitioning schemes assuming you'd want them to take over the entire disk. GhostBSD's *GBI* installer uses the *pc-sysinstall* back-end developed by TrueOS.



➤ NetBSD's installation is the most esoteric of all. For instance, the disk partitioning step involves selecting the bootblocks that you want to install.

Verdict

DragonFly BSD
★★★★★
GhostBSD
★★★★★
MidnightBSD
★★★★★
NetBSD
★★★★★
TrueOS
★★★★★

» All the installers choose quite sensible defaults to ease installation.

System management

How do you administer your realm?

Besides an easy to use installer, the availability of custom administration tools often help persuade users to move to a different computing environment. All the BSD OSes in the Roundup have custom binary package management tools to make it painless to flesh out installs.

MidnightBSD uses the home-brewed *mports* package management system that works in a similar way to *APT* to search, install and upgrade applications. Currently, it indexes about 2,600 packages for i386 and about 1,800 for amd64 architecture. Similarly, DragonFly BSD uses FreeBSD *Ports* as a base for its own ports collection (called Delta Ports), and you can also install packages using FreeBSD's *pkgng*.

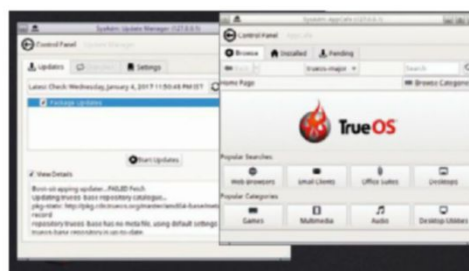
The default NetBSD install is also minimal but you'll first have to set up its binary package manager before using it to install a desktop. Also, some BSDs, such as DragonFly BSD, use a

configuration tool that runs at the end of the install to set up key aspects of the system such as the networking options.

While the package management systems of these three BSD OSes give access to all kinds of software they all lack a graphical package manager.

If you're spoilt by graphical config tools on modern Linux distros, you won't get very far, especially with NetBSD as everything has to be set up by hand.

GhostBSD uses the standard system configuration tools that ship with the default desktop as well as *Dconf* editor for fiddling with *GTK* settings. The OS relies on the *OctoPkg* front-end for its *pkgng* package management system and also makes it fairly easy to update both the system as well as installed apps.



➤ TrueOS makes it fairly easy to add apps and keep the system updated using the intuitive *AppCafe* and *Update Manager* utilities.

TrueOS trumps the others in the number of custom tools. Its Control Panel helps you manage different aspects of your installation, such as adding new users, configuring network connections, setting up the firewall and more. Then there's the backup tool, *Life Preserver*, which can sync to a remote FreeNAS system securely, using *rsync* and SSH. The OS uses FreeBSD's ports and also publishes packages in its own PBI file format, which you can install via its *AppCafe* graphical package manager.

Verdict

DragonFly BSD
★★★★★
GhostBSD
★★★★★
MidnightBSD
★★★★★
NetBSD
★★★★★
TrueOS
★★★★★

» TrueOS beats the others by offering many custom graphical admin utilities.

Best Desktop BSD

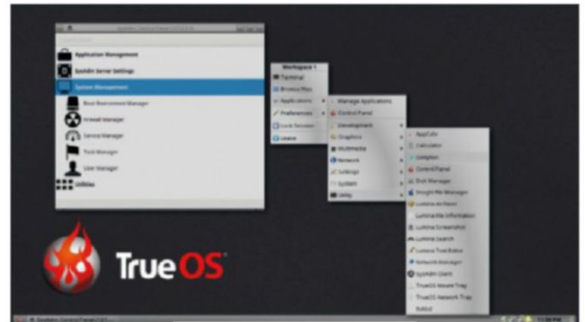
The verdict

Unlike some of the other Roundups, it's easier to recommend a BSD for a desktop user. While NetBSD, MidnightBSD and DragonFlyBSD can all be used on a desktop, you'll have to build them from a barebones command-line interface by pulling in packages using their binary package managers. The process isn't as tedious as it sounds, but these three are more suitable to the DIYers who are used to the ways of geekier Linux distros, such as Arch Linux and Slackware.

Between them, these three BSDs have advantages that make them suitable for certain situations. For example, if you want to run a BSD-based desktop on your microwave you'll have a better chance with NetBSD. Similarly, if your day job involves setting up virtual hosting environments on shared servers, you'll find it easier to tune DragonFlyBSD into a desktop OS for everyday use.

But if you're looking for point and click simplicity, the real desktop options are GhostBSD and TrueOS. These projects are as close as BSD can get to Linux on the desktop. Both of these projects offer the same functionality, convenience and applications that Linux users tend to expect. While TrueOS, with its custom management tools, is a better-equipped desktop OS than GhostBSD, it's more restricted than the latter. For starters, if you are planning to use BSD on a 32-bit machine, you'll have to stick with GhostBSD as TrueOS doesn't put out ISO images for that architecture. Second, the default desktop on TrueOS is Lumina while GhostBSD uses the more familiar MATE and Xfce desktops.

However, we've given first place to TrueOS as besides not running on 32-bit hardware, TrueOS can do everything that GhostBSD can and then some. You can easily replace TrueOS's default Lumina desktop with a more



» TrueOS isn't just a wonderful BSD for the desktop; it can also teach a thing or two to some Linux distros as well.

familiar one. It also has a much larger infrastructure, team and budget to sustain and support the project. This is evident from the slew of custom graphical admin tools. TrueOS is one of the few BSDs that uses an intuitive graphical installer that does a nice job of not scaring away first timers. It also stands out for having a graphical package manager and a large number of packages to flesh out the default installation. All things considered, GhostBSD and TrueOS are your best bets to dabble with BSD and the two least likely to throw surprises at you.

1st**TrueOS** ★★★★★Web: www.trueos.org Licence: BSD licences Version: 2016-12-27

» BSD's best bet to challenge Linux on the desktop.

4th**NetBSD** ★★★☆☆Web: www.netbsd.org Licence: 2-clause BSD Version: 7.0.2

» Easily transforms into a desktop on all kinds of hardware.

2nd**GhostBSD** ★★★★★Web: www.ghostbsd.org Licence: Simplified BSD Version: 10.3

» The only real BSD option for desktop users with 32-bit hardware.

5th**MidnightBSD** ★★★☆☆Web: www.midnightbsd.org Licence: BSD licences Version: 0.8.0

» Aims to be convenient for desktop use, but isn't there yet.

3rd**DragonFly BSD** ★★★☆☆Web: www.dragonflybsd.org Licence: Modified BSD Version: 4.6

» Though it doesn't ignore desktop users, it's best for server deployments.

Over to you...

Feeling adventurous enough to try a BSD desktop in the new year? Share your experience with us at lxf.letters@futurenet.com

Also consider...

It really shouldn't be a surprise that there's a long list of BSD-based OSes that are under active development and can be transformed into a desktop for everyday use. There's Debian GNU/kFreeBSD, which swaps out the Linux kernel and uses the FreeBSD kernel instead together with GNU-based userland utilities and glibc. It's developed by the Debian

project, which maintains two ports based on the FreeBSD kernel, kfreebsd-i386 and kfreebsd-amd64. Similarly, there's Gentoo/FreeBSD and Gentoo/OpenBSD, that aim to port unique Gentoo features such as the *Portage* package manager to FreeBSD.

You can use OpenBSD itself as an everyday desktop. OpenBSD's sole purpose is to be the

most secure OS, which usually puts it at odds with convenience. While it's certainly possible to use OpenBSD as a desktop platform, this isn't a top priority for its developers. Then there are a slew of BSD-based projects that cater to a niche segment, e.g. BlackBSD is a NetBSD-based live CD that ships with security tools on top of the *Fluxbox* window manager. **LXF**

Subscribe to

LINUX FORMAT

Get into Linux today!

Choose the perfect package for you!

» GET THE PRINT EDITION



» Every issue comes with a 4GB DVD packed full of the hottest distros, apps, games and loads more!

Only £14

Every 3 months by Direct Debit

» GET THE DIGITAL EDITION



» The cheapest way to get *Linux Format*. Instant access on your iPad, iPhone and Android device.

Only £11.25

Every 3 months by Direct Debit

» GET THE BUNDLE DEAL

Get both the print & digital editions for one low price!

£19

Every 3 months by Direct Debit

**SAVE
49%**



» **PLUS:** Exclusive access to the *Linux* Format subs area—1,000s of DRM-free issues, tutorials, features and reviews.

Subscribe online today...

myfavouritemagazines.co.uk/LINsubs17

Or Call: 0344 848 2852

Prices and savings quoted are compared to buying full-priced UK print and digital issues. You will receive 13 issues in a year. You can write to us or call us to cancel your subscription within 14 days of purchase. Your subscription is for the minimum term specified and will expire at the end of the current term. Payment is non-refundable after the 14 day cancellation period unless exceptional circumstances apply.

Your statutory rights are not affected. Prices correct at time of print and subject to change. UK calls will cost the same as other standard fixed line numbers (starting 01 or 02) and are included as part of any inclusive or free minutes allowances (if offered by your phone tariff).

For full terms and conditions please visit: bit.ly/magtandc. Offer ends 31/03/2017

Escape Google (and friends)

Jonni Bidwell wants you to take back your data from the internet megacorps and he knows some open source services that will help replace these proprietary demons.



Abandoning (or at least reducing your dependency upon) proprietary services needn't be about being some weird information vegan (to use a phrase from Cory Doctorow). It should be about a genuine concern for the increasing amount of data we're handing over to companies, how they are using that data to profile our online behaviour and how those profiles and relationships are used to bombard us with targeted advertising.

Microsoft and Google know a great deal about users through their respective smartphone OSes, Facebook and Twitter use

the ubiquitous Like or Retweet buttons to track users' behaviour all over the web.

And this is a shame, because there are plenty of alternative services that don't involve divulging your personal data to a

“There are plenty of alternative services that don't involve divulging your personal data.”

corporation that will use it to get better at making you buy things or offer it up to whatever government agency asks for it. In fact, by hosting these things yourself (and we'll show you how to host your own

Dropbox-killing *Nextcloud* instance) you can be the master of your own data. The EFF (Electronic Frontier Foundation) has a slogan, “There is no cloud, just other people's computers” and many people would do well to heed it.

We'll present a couple of alternative services, some of them run on other people's computers, but those people's motives are entirely honourable. We cordially invite

you to encourage your friends and followers to these greener pastures. But be prepared for some of them not to—you may want to keep your Facebook account alive if you have FOMO (fear of missing out).

Easy ways out

It's very easy to stop giving your web and map searches to the Google.

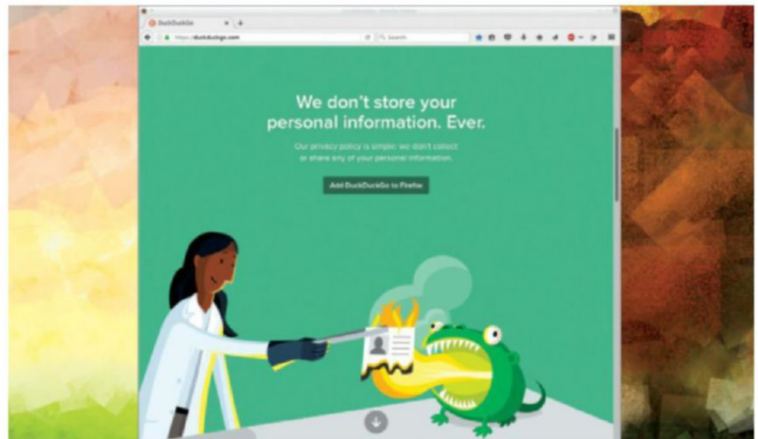
More often than not, quitting an online service altogether will be an inconvenient, isolating experience. By abandoning one network or another we may lose the ability to contact others or be forced to do so over old fashioned channels like email or have to meet them in person.

There may be equivalent (and probably better) services available, but that point is moot if the people with whom you want to commune are not using them. So rather than completely severing ties with a given proprietary service and exclusively using the alternative in general, it's easier to use the more wholesome alternative service in parallel with the one you'd rather move away from, all the while dropping subtle hints for others to do the same. For example, it's easy to have multiple email accounts, and change the reply-to addresses on those that live in proprietaryville, using ever more aggressive passive-aggressive tones at those that continue to use them.

That said there are some services that can easily be abandoned. Google introduced a unitary privacy policy back in 2012, which means that your interactions with all its services are all tied to a single Google ID. This certainly makes for a simpler privacy policy, but also allows for much more effective data mining. If you use your Google ID with *Google*

“Your interactions with all Google’s services are tied to a single Google ID.”

Chrome browser or even if you just keep your Google account logged in, e.g. through an always open Gmail tab, then all your Google and Map searches are stored and cross-referenced with the rest of your Google profile so that appropriate ads can be foisted upon you. These search data are easily found on the Dashboard in the My Account page, as are audio snippets of any uses of ‘OK Google’ functionality from an Android phone. An individual’s search queries can be very telling, given that they might search for medical conditions, bankruptcy advice or directions to their hotel. While these



» If there's one thing we can all agree on, it's that we need more fire-breathing creatures in open source. And possibly give DuckDuckGo a try.

details are never sold to anyone, the fact that they are stored at all is a concern.

You might not want to stop using all of Google's services, but it's easy to switch to a more privacy conscious search engine. One such engine that has become very popular is DuckDuckGo (<https://duckduckgo.com>) which is now serving over 12 million requests every day. DDG has soared in popularity thanks to its policy of not collecting or sharing personal information. Not only is DuckDuckGo easy to add to your favourite web browser's search box, but if

you're unsatisfied with its search results then it can easily provide Google results instead—just prefix your search query with `!g`. This will be passed to Google's encrypted search page, which won't pass unencrypted referrer data to any ads or search results you click on, but it is still Google, so that data is passed to HTTPS sites, and will be stored by Google (you can't get away from Google by continuing to use it). A number of other providers can be searched via the `!` syntax. These are known as bangs, and a list of over 9,000 providers are available at <https://duckduckgo.com/bang>. »

OpenStreetMap – Get thy bearings

Google Maps is an extremely useful service, especially now that it's good at planning journeys using public transport. But letting the search giant know where you're going and when you're going to be there is just spoon-feeding it data to add to your profile and allows it to get marginally richer by sending spatiotemporally relevant ads our way.

Much better would be to support the community oriented OpenStreetMap (OSM) project. It's supported by a not-for-profit foundation which you should join for a paltry

£15. OSM has been around since 2004, when Steve Coast decided to take a stand against proprietary map data in the UK and felt a Wikipedia-style crowd-sourced approach might work. Since then map data has been collected by volunteer surveyors and GIS enthusiasts or contributed by various agencies, most notably Automotive Navigation Data, which provided road data for the Netherlands, China and India. Yahoo! also provided satellite imagery from which vector maps could be freely produced. The results are impressive—certainly there are

locations that require attention [you could help!—Ed], but most cities in the developed world now have reasonable OSM coverage.

Gnome Maps is powered by OSM and is – after a hiccup with tiling data last year – a slick application for finding your way around. OSM efforts have proven invaluable in crisis situations too. OSM has a dedicated Humanitarian Team who have supported disaster-relief efforts following the Haiti earthquake, the West African Ebola outbreak and Typhoon Haiyan in the Philippines.

The GNU AGPL

As more and more networked services appeared, it became obvious that we needed a FOSS licence for the online generation.

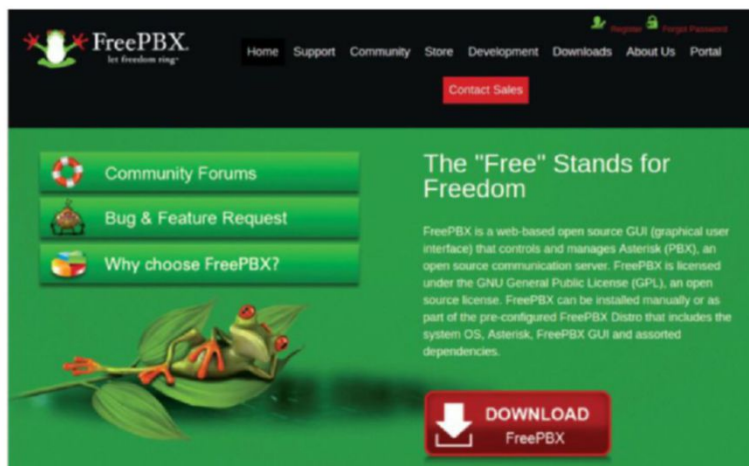
» **T**he original GPL (General Public License) was written in 1989, long before most people started to worry that a couple of decades down the line we'd all be connecting to remote services to which vague and nebulous 'societal pressures' would coerce us into handing over our life stories, cat pictures and arm's-length tales of self promotion. The GPL and its 1991 sequel, both deal only with code that's distributed to directly to users and run locally on their machines.

Where the GPL definitely doesn't extend to is code that's not distributed to users; code that runs remotely and sends responses back to users' machines. So this covers all the server-side languages and just about any cloudy online service that the millennial sat next to you on public transport might be using. By using these services, where all the interesting code is run remotely, the copyleft portion of the GPL isn't triggered and the user is not entitled to a copy of the source code. Thus it is perfectly possible for an organisation to take some GPL-licensed code, modify it to run as a service, and not be compelled to distribute their modifications.

Today we're quite familiar with the notion of Software (or indeed pretty much any noun you like) as a Service (SaaS), but back in the day this was referred to as the Application Service Provider (ASP) loophole. At the turn of the millennium, discussions took place between Richard Stallman, technology pioneer Henry Poole as well as the FSF's Bradley Kuhn and Eben Moglen around how to close the ASP loophole. Poole's company, Affero Inc., was set up in 2001 and Poole wanted to be able to license the code for Affero's web services, so that others could create derivative works so long as they were done in the spirit of copyleft. Kuhn proposed adding an appropriate clause to GPL v2 and together with Moglen came up with the text, which in lay speak affords network users the same right to source code as if the software were running locally. The resulting derivative licence was approved by the FSF and published by Affero in 2002, known as the Affero General Public License (AGPL v1).

Three years later, by which time demons such as Digital Rights Management (DRM), tivo-isation and software patents had reared their ugly heads, work was underway on a new GPL version. Early drafts of GPL v3 included an Affero-like clause, but to ease administrative burden this was not included in the final text, published in 2007. Instead, the FSF published a new licence, which was essentially GPLv3 with the Affero provision, and this licence was dubbed the GNU AGPL v3 (in the name of matching version numbers). This is the latest version of the licence and is recommended by the FSF "for any software which will commonly be run over a network." In the name of completeness, there is also a transitional AGPL v2, which enables AGPL v1-licensed code to be redistributed under AGPL v3.

Why does any of this matter? Well it enables developers to create useful, open source network programs without risk of unscrupulous actors pinching their code and using it for profit. As a result, we have some fantastic FOSS alternatives to proprietary online services. Some of these we'll cover here, and others we'd encourage you to investigate for yourself. The most well-known to readers will be Nextcloud. (See p34, the 'light side' in last year's schism among the ownCloud ranks.)



» **FreePBX is an AGPL-licensed project for controlling the Asterisk private branch exchange (PBX), which in turn manages PSTN and VoIP calls.**

AGPL in the wild

Canonical's project hosting site and bug-tracking Launchpad has been around since the early Ubuntu days. Many Canonical projects, including Launchpad itself, are hosted here. Initially Mark Shuttleworth rejected calls to open source the project. However, the community's chagrin persisted and the software was eventually released under AGPL v3 in 2009. The site is now more popular than ever, and we haven't seen a proliferation of almost-but-not-quite compatible Launchpad clones in the wild, so it's fair to say that this was a good decision.

It is home to some high-profile projects, such as Linux Mint, MariaDB and elementary OS.

Massive open online course (MOOC) provider EdX has been entirely open source since 2013, allowing anyone from anywhere to study anything, in many cases for free. Thanks to the platform's licensing, educational institutions around the world are now offering their online initiatives. The IMF uses the edX platform to offer economics and finance courses. The Internet Archive's Open Library project aims to create "one web page for every book ever

published" and provides access to many public domain texts, as well as a system for borrowing eBooks. This, together with efforts such as Project Gutenberg, provide a nice alternative to Google and Amazon's bookish offerings.

One of the most promising projects in the field of artificial intelligence is the OpenCog framework, licensed under the GNU AGPL. Whether or not you believe Ray Kurzweil's prophecies about the coming of 'the singularity', it's good to know that there are free platforms on which we can architect our robotic demise.



Escape Facebook

If you can cut the ties to the Zuckerberg, and better yet convince some friends to join you, the world will be a better place.

Facebook is a lot of things to a lot of people. From our cynical outlook at *Linux Format Towers*, it appears to be a platform on which users provide all the content and in return are rewarded with targeted adverts. For some, it has engendered a most unhealthy need for a constant stream of validation from their peers, giving rise to a whole commerce of the blue thumbs, red hearts and yellow smiley faces.

“Diaspora* is a decentralised service where anyone can set up and run their own instance.”

Woe betide they that don't pay due deference to the pictures of the dog's breakfast this morning, or Mr and Mrs Jones's feigned expressions of joy in their holiday snaps. Or you can just while away the hours scrolling and lurking and making yourself feel bad. Many brave souls have abandoned Facebook and still seem to be doing okay, a few missed parties notwithstanding. Perhaps now is the time to take the plunge and join them?

The Diaspora (officially written 'diaspora*') social network (<https://diasporafoundation.org>) was officially launched in 2011. Unlike Facebook, it's a decentralised service where anyone can set up and run their own instance of the software, referred to as a 'pod'. Users can also create an account on an existing pod, and pods are can communicate with one another allowing for federated access across the whole network. This approach led to problems when fundamentalist groups, evicted from Twitter, moved their propaganda operations to Diaspora. Their accounts were stored on pods administered by individuals sympathetic to their causes, and so those 'podmins' generally ignored requests to remove offensive material. This is unfortunate, but once this kind of material is identified, community members can notify ISPs and, if necessary, local law enforcement, so that appropriate action can be taken.

Diaspora runs as a Ruby on Rails application and setting up your own pod is a slightly involved process. However, instructions for setting it up with common distros, web servers and database back-ends can be found on the wiki. If you just want to try out the network, join one of the public pods helpfully listed at <https://podupti.me>.

Another decentralised social network is Friendica. Friendica can integrate with Diaspora, GNU Social and indeed Facebook and Twitter. Again, installation on your own server is a little involved (Friendica is a PHP application and requires the mail() function to work, so a working email gateway or server is required), but a *Virtualbox* image is available which can be run on a local machine. With a bridged network, dynamic DNS name and textbook port-forwarding this will work just like a fully fledged server, albeit with vastly reduced capacity.



» GNU Social is a federated microblogging platform for everyone. There are bots that capture many well-known Twitter accounts.

You can't stop the signal

WhatsApp last year rolled out end-to-end encryption across its network, an effort which we applauded. It takes a great many smarts to get this working seamlessly and transparently, and it's a shame when news outlets start peddling stories about bugs or backdoors which aren't really there. That isn't why we should worry about WhatsApp, we should worry because its network is owned by Facebook. While data sharing for advertising

purposes has been 'paused' in the UK and Europe, there has been no indication of how long this reprieve will last.

Our recommendation for secure messaging is the amazing Signal, by Open Whisper Systems. The underlying Signal Protocol is the same as that used by Whatsapp, but Signal is open source (see <https://github.com/WhisperSystems>) and not owned by Facebook. The app is easy to install from Google or Apple

app stores, and a *Chrome* app is available for desktop users, but since accounts are tied to phone numbers it's necessary to install a mobile version first. Signal is cosmetically similar to WhatsApp, and since all the fancy key generation and exchanges take place behind the scenes, there's nothing to configure once it's installed. Any contacts that are using Signal already will show up and as you persuade others to join the network they will appear, too.

Free your email

Hosting an email server no longer requires blood sacrifices and a resident dark wizard, but there are mages that will do it for a fee.

» **A**bout a decade ago, Google opened up its hitherto invitation only webmail service to the masses and the masses came in their droves. Seduced by the free 2GB of storage, the clean interface and, most importantly, the fact that it saved your drafts pretty often, so when your connection was interrupted and the page reloaded your polite missive was mostly intact.

This was made possible by some impressive JavaScript voodoo which eventually became the well-known AJAX. Microsoft and Yahoo were forced to up their webmail offerings and soon most everyone was using one of these services for their primary email. The price of this convenience was that one's search provider was now one's email provider, and they had no scruples about scouring one's messages and contacts for clues about what to advertise on one's search results.

We've shied away from covering setting up your own email server guides in the past for a number of reasons. One is that there's several different components involved and these all deserve quite a lot of words to set up in a sane manner. Another reason is that other email servers can be pretty untrusting and with good reason—there's a huge number of dodgy email servers running on compromised machines that

are spewing spam messages 24/7. To combat this, a server reputation system has been developed, but unfortunately in some situations having no reputation is not much better than having a bad one. Until mail is accepted successfully and received by another server, the sending server's reputation will not improve. So there is potential for a catch-22 situation. Worse, if your email server is compromised then its IP will very likely be blacklisted for a Very Long Time.

Be that as it may, it almost worked for Hillary Clinton and there are now some great complete email solutions including Mail-in-a-Box, iRedMail and Mailcow (<http://mailcow.email>). Mailcow combines *Postfix*, *Dovecot*, *Spamassassin*, *ClamAV*, *OpenDKIM*, the rather splendid *Roundcube* webmail interface, and the database and web server of your choosing, into one easy to install package. It's all very well running your own email server, but what happens when it breaks? It's alright if it was just a hobby project – just pick up the pieces and try again with a better backup or security strategy – but for your primary email account why not avoid that burden and use a different email provider?

Protonmail is a privacy conscious email provider incorporated in Switzerland (many of its staff worked at CERN). It offers free and paid-for email services and since August 2015 its entire codebase is open source. Using its webmail interface, emails between Protonmail users are end-to-end encrypted – users have two passwords, one for authenticating with the service and a mailbox password which is used to decrypt a private key – so emails sent and stored on the service can't be accessed by ProtonMail or anyone else. ProtonMail's web interface can also send asymmetrically encrypted email to outside addresses, directing the recipient to a web page where they must enter a preshared private key (sent from the sender) to view the email. The service doesn't work with traditional IMAP, SMTP or POP3 clients, but mobile apps and Two Factor Authentication (2FA) are available.

An alternative service is FastMail, and its premium service grants the ability to use your own domain. Data at rest is stored encrypted and tools are available for migrating email from IMAP accounts. Hushmail is also worth investigating as, like Protonmail, it offers a free service with transparent encryption between users, and, like FastMail, its paid-for services can work with your own domain name.



mailcow mailserver suite

> News and notes

29. Dec 2016 - mailcow: dockerized

11 containers and 4 volumes combine to a Docker conform mailcow installation.

mailcow: dockerized will be a rolling release in the master branch @ GitHub. There will be a dev branch soon, which should not be used.

Learn more about mailcow: dockerized below, please visit our board @ forum.mailcow.email or visit us on IRC @ Freenode in #mailcow

[Link to GitHub](#)
[Link to Docs](#)

» Ruminants have multiple stomach compartments and our bovine messenger is now available, in eleven easy to digest docker containers.

DIY email service

If you're interested in hosting your own services and your home internet connection has the bandwidth to support them (and your ISP permits you to do such things), then it's entirely possible to host these things at home. However, a Virtual Private Server may be a better option—inviting the internet onto your home network should only be done with security precautions in

place. A suitable starter VPS can be had for around £5/mo. If you're setting up your own email server then you'll need to pay a registrar for a domain name and a DNS provider. For web-facing services (such as *Nextcloud* which we'll cover later), a subdomain will be sufficient, and this can be arranged through a dynamic DNS provider such as DuckDNS. For web

services where personal information or passwords are transmitted, HTTPS is essential. This authenticates that the server is who it claims to be and that all data between it and the client is encrypted. Thanks to Let's Encrypt (see *Beep Bop, I'm a Certbot box, right*), free SSL certificates are available to all and can be used with dynamic DNS services too.

The Lavabit story

Ladar Levison fought to protect the privacy of his clients, but when one of those clients leaked damning state secrets things got ugly.

We don't often cover paid-for services in LXF, but we really do approve of the one's we've mentioned here. Even a virtual host lives on physical hardware and that needs to be maintained and protected from all manner of digital malfeasance. Email is particularly tricky to do securely, especially if your threat model accounts for black-suited agents going to your host with scary looking paperwork and removing hard drives.

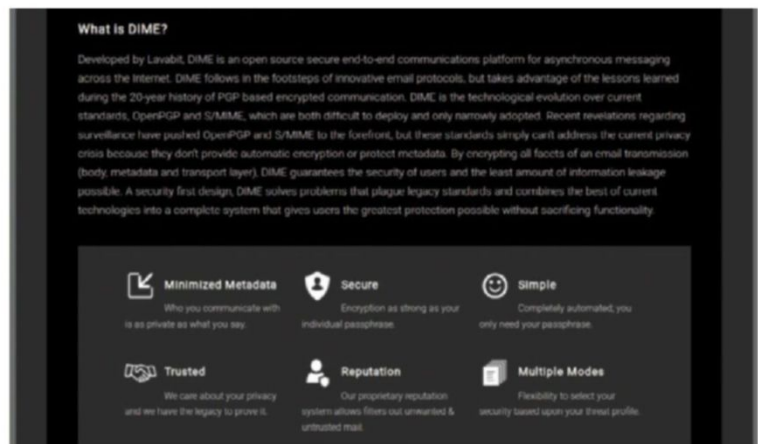
An ongoing case brought by the US government two years ago seeks to compel Microsoft to hand over emails belonging to one of its Live Mail customers. The data is stored in a Dublin data centre and Microsoft has argued (successfully so far) that US law doesn't have jurisdiction there. While its refusal may not have helped the defendant in this case, the court's decision goes some way towards checking intelligence agencies already widespread powers of investigation.

We can applaud Microsoft for its stance here, just like we could applaud Apple for its refusal to obey court orders, filed under the obscure All Writs Act of 1789, to assist in the extraction of data from iPhones, but it's worth noting that both of these giants, as well as Google, Yahoo!, Facebook and AOL were named as being complicit in NSA's data-slurping PRISM programme. This programme was revealed, along with all kinds of government-shaming material by Edward Snowden in 2013.

“Our hero stuck by his guns and for his principles was slapped with a contempt of court charge.”

Snowden, it was later revealed, had been using a secure email provider called Lavabit, run by Ladar Levison, to communicate the offending material to journalists. When the feds learned of this, they paid Levison a visit. With legally compelling paperwork in hand, agents insisted that Levison install a surveillance device which would intercept Snowden's emails as they left the Lavabit network. This 'pen register' he claims to have agreed to, but later the feds wanted more.

Lavabit had an encrypted storage feature and the law wanted to know what Snowden had stored there, so it



» Lavabit launched a new email service based on the Dark Internet Mail Environment (DIME) platform at the beginning of the year..

demanding Lavabit's private key. This would grant the government the ability to read not just Snowden's emails, but those of Lavabit's 410,000 other customers.

After a protracted series of events in which papers, subpoenas and search warrants were served, Levison and his legal counsel found themselves in a Kafkaesque situation. An order of contempt was issued against Levison, but was done so without a hearing. Thus Levison was unable to defend himself or object. And without any objection, the appellate court upheld the contempt charge. Levison shutdown Lavabit in August 2013, saying to do otherwise would mean being "complicit in crimes against the American people."

Not only is Levison a man of principle, but also a hombre con cojones. On 1 August 2013, a court ordered him to hand over the SSL keys the next day. Levison did this, but in the form of eleven pages of unscannable 4-point type. This, the court did not approve of and demanded the key be handed over in "an industry-standard electronic format". Our hero stuck by his guns and for his principles was slapped with a contempt of court charge. »

Beep Bop, I'm Certbot

Whatever services you're running, it's never a bad idea to run them over HTTPS, and thanks to Let's Encrypt this can now be arranged for free. It's unlikely that you will find yourself, like Levison, slapped with an order from a secret court to reveal the private keys.

Managing certificates can be a tedious affair, but the EFF has come to our rescue with certbot, a simple tool for fetching and deploying SSL certificates. Given only a domain name

(dynamic DNS ones are fine) and optionally an email address (in case revocation is required) it can generate a certificate and have Let's Encrypt signed it so that it will be accepted by all major browsers. For Apache users, it can reconfigure things to start serving with the newly hatched certificate immediately and set up a sane SSL configuration for you.

CertBot (<https://certbot.eff.org>) is in the Ubuntu repos or via the Backports repos for

Debian Jessie. For the latter, install it with:

```
$ sudo apt-get install certbot -t jessie-backports
or python-certbot-apache if you want the
Apache plugin. See the website for other distros
and web servers. To use the fire and forget
mechanism is a simple case of $ certbot
--apache or use the certonly option if you'd
rather make the changes by hand. Certbot will
even set up a cron job to update certificates
before they expire.
```

Run your own Nextcloud

Remote storage silos are a dime a dozen. But be wary of their privacy policies, we explain how to set up your own cloud.

» **O**nline storage services such as Dropbox offer a convenient option for accessing and sharing data anywhere on the planet. Yet the conveniences comes at a cost, and the very idea of transferring our files to a remote server, outside of our jurisdiction, seems rather strange in the post-Snowden era. This is where *Nextcloud* steps in. It offers all the convenience of an omnipresent storage service while keeping you in charge of your private data.

The open source data sharing server is brimming with features for both home and large-scale enterprise users. With *Nextcloud*, you can store, sync and share not just your data but your contacts and calendars. It also boasts of advanced features such as single sign-on capability, theming for custom branding, custom password policy, secure WebRTC conferencing, Collabora Online Office integration and more. If that's not enough, in addition to the core functions you also get a host of additional useful add-ons.

Paint the sky

You'll have to lay the foundation before you can install *Nextcloud*. We'll set up *Nextcloud* on top of an Ubuntu Server 16.xx installation and begin by making sure our installation is up to date with `sudo apt update && sudo apt upgrade`.

We'll then fetch and install the individual components that make up the popular LAMP stacks. First up is the Apache web server which can be installed with `sudo apt install apache2 apache2-utils`. Next up is *MariaDB* which is a drop-in replacement for *MySQL*. Install it with `sudo apt install mariadb-server mariadb-client`. Straight after it's installed, run *MariaDB*'s post installation security script with `sudo mysql_secure_installation`. The script takes you through a small command-line wizard to help you setup a password for the database server's root user and setup some other defaults to harden the database installation. Then comes PHP which you can fetch along with all the required modules with:

```
$ sudo apt install libapache2-mod-php7.0 php7.0-json
```

```
php7.0-curl php7.0-gd php7.0-mysql php7.0-xml php7.0-zip
php7.0-intl php7.0-mcrypt php7.0-imagick php7.0-mbstring
```

By default, PHP has defined very conservative limits which will prevent you from uploading large files into your *Nextcloud* server. The following commands will increase the PHP memory limit to 512MB, and take the upload and individual post size to 250MB:

```
$ sed -i "s/memory_limit = .*/memory_limit = 512M/" /etc/
php/7.0/fpm/php.ini
$ sed -i "s/upload_max_filesize = .*/upload_max_filesize =
250M/" /etc/php/7.0/fpm/php.ini
$ sed -i "s/post_max_size = .*/post_max_size = 250M/" /etc/
php/7.0/fpm/php.ini
```

Last, ensure that the PHP module is loaded in Apache with `sudo a2enmod php7.0` before you restart the web server with `sudo systemctl restart apache2`.

Next, we'll create a database and a *Nextcloud* user in *MariaDB*. Log into *MariaDB* database server with `$ mysql -u root -p`. After authenticating with the password you specified while securing *MariaDB*, you can create a database named *nextcloud* with `> create database nextcloud;`. Similarly, you can create a user for administering this database with:

```
> create user nextcloudadmin@localhost identified by
'a-password';
```

Remember to replace *a-password* with your preferred password. Finally, grant this user all the privileges on the freshly minted *Nextcloud* database with:

```
> grant all privileges on nextcloud.* to nextcloudadmin@
localhost identified by 'a-password';
```

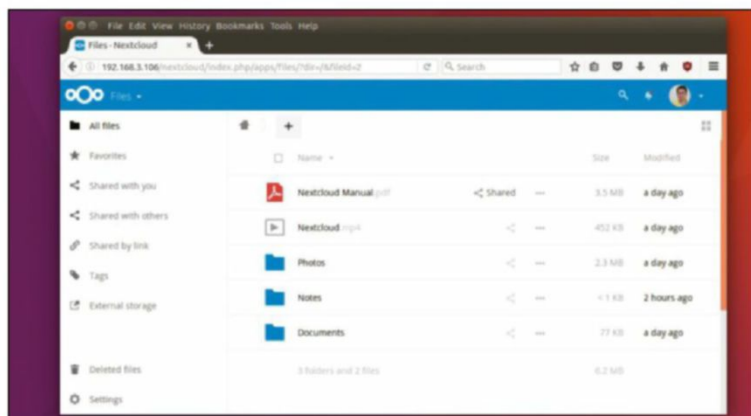
```
Bring the changes into effect and exit the MySQL prompt:
> flush privileges;
> exit;
```

We'll also enable the binary log for *MariaDB* which will contain a record of all the changes to the database. Open *MariaDB*'s configuration file in a text editor with `sudo nano /etc/mysql/mariadb.conf.d/50-server.cnf` and enter the following lines under the `[mysqld]` section:

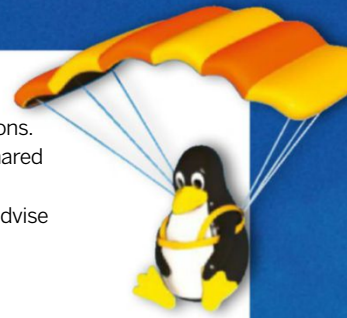
```
log-bin = /var/log/mysql/mariadb-bin
log-bin-index = /var/log/mysql/mariadb-bin.index
binlog_format = mixed
```

Save and close the file when you're done. Then reload *MariaDB* service with `sudo systemctl reload mysql`. Similarly, you'll also have to make some tweaks to the Apache web server. *Nextcloud* needs several modules to function correctly. Enable them with the `a2enmod rewrite` and `a2enmod headers` commands. It's recommended to install the *env*, *dir* and *mime* modules too.

Also while you can use *Nextcloud* over plain HTTP, the *Nextcloud* developers strongly encourage the use of SSL/TLS to encrypt all server traffic and to protect user's logins and data in transit. Apache installed under Ubuntu already comes equipped with a simple self-signed certificate. All you have to do is to enable the SSL module and the default site and accept the use of the self-signed certificate. (Alternatively, see



» Unlike its progenitor, there is only one open source version of *Nextcloud* and the developers plans to generate revenue via support and consulting services.



the Beeb bop, I'm a CertBot box, p33, to fetch and deploy certificates yourself):

```
$ sudo a2enmod ssl
$ sudo a2ensite default-ssl
```

When you are done, restart the Apache server to load the modules with `sudo systemctl restart apache2`.

In the clouds

Now that we've laid the groundwork for *Nextcloud*, let's fetch and install the server. Head to www.nextcloud.com/install and grab the latest version (which is v11.0.1 at present):

```
$ wget -c https://download.nextcloud.com/server/releases/
nextcloud-11.0.1.tar.bz2
$ tar xvf nextcloud-11.0.1.tar.bz2
```

Deflating the archive will create a new directory named **nextcloud** in the current working directory. Copy the new directory and all of its content to the document root of the Apache server with `sudo cp -r nextcloud /var/www/`. Then hand over the control of the directory to the Apache user (www-data) with `sudo chown www-data:www-data /var/www/nextcloud/ -R`.

We'll install and access *Nextcloud* from under its own directory by creating a configuration file with `sudo nano /etc/apache2/sites-available/nextcloud.conf` and the following:

```
Alias /nextcloud /var/www/nextcloud/
```

```
<Directory /var/www/nextcloud/>
Options +FollowSymLinks
AllowOverride All
```

```
<IfModule mod_dav.c>
Dav off
</IfModule>
```

```
SetEnv HOME /var/www/nextcloud
SetEnv HTTP_HOME /var/www/nextcloud
</Directory>
```

Save the file and bring *Nextcloud* online with:

```
$ sudo ln -s /etc/apache2/sites-available/nextcloud.conf /etc/
apache2/sites-enabled/nextcloud.conf
```

That's the command-line stuff taken care of. Now fire up a web browser on any computer on the network and head to <https://192.168.3.106/nextcloud>. Replace **192.168.3.106** with the IP address or domain name of the server that you've deployed *Nextcloud* on.

Since this is the first time that you're interacting with *Nextcloud*, you'll be asked to create an admin account. Enter the username and password for the *Nextcloud* administrator in the space provided. Then scroll down and expand the

Storage & database pull-down menu to reveal more options. The **data** folder is where *Nextcloud* will house the files shared by the users. Although it'll already be populated with a location, for security reasons the *Nextcloud* developers advise that it's better to place the data directory outside the *Nextcloud* **root** directory, such as **/var/www/data**.

You're next prompted for several details about the database server. By default *Nextcloud* uses the *SQLite* database which is adequate for smaller installations. However, we've already set up the industry-standard *MariaDB* which can handle all sorts of loads. Use the textboxes to enter the username and password for the user we created earlier to manage the nextcloud database. Then press the 'Finish setup' button to enable *Nextcloud* to connect to the database and create the appropriate structure for the installation.

That's it, your *Nextcloud* server is up and running. You'll now be taken to *Nextcloud*'s dashboard. While you can start using the server to upload and download files straight away, let's take a moment to get the house in order.

For starters, roll-down the menu next to your username in the top-right corner and click the 'Personal' link. Here you can review and change several settings for your account, such as the password and display name. It also lists the groups you are part of. If your *Nextcloud* deployment is going to be used by multiple people, it's advisable to organise users into

“An omnipresent storage service while keeping you in charge of your private data.”

different groups. To do this, select the Users option from the pull-down menu. You can then use the forms on the page to create groups and users. While adding users, you can also restrict their storage space and even mark certain users as administrators of particular groups.

You're now all set to upload data into your *Nextcloud* server. After you've logged in, you are dropped in the Files section. The interface is very intuitive and straightforward. To upload a file, click on the '+' button and choose 'Upload' from the drop-down menu. To organise files into folders, click on the '+' button and select the 'Folder' option. If you've uploaded a file in a format that *Nextcloud* understands, you can click on its name to view and edit the file. *Nextcloud* can visualise the data it houses in different views. For example, click on the Files pull-down menu in the top-left corner of the interface, and select the Gallery option. This view helps you view images in your cloud by filtering out all other types of content. »

Cloud control

We've looked at setting up and using a default *Nextcloud* instance. But as the admin you can tinker with several settings to tailor *Nextcloud* to your requirements.

To access these settings, roll down the menu next to your username and select the Admin option. This takes you to a page that lists several settings that affect the entire *Nextcloud* installation grouped under various different

heads such as Server settings, Server info, Usage report and more. The Server info option is different from the others in that instead of helping you tweak any settings it only visualises various details about the *Nextcloud* server such as the load on the CPU, memory usage and more. Head to the Sharing section to configure the policy for sharing files on the server. Here you can toggle options to force users to set a

password on all public shares, set a default expiry date for all public shares and restrict members of share files only with others users in their group etc. You can configure the *Nextcloud* server to send out emails for various types of notifications and password resets from the Additional settings section. This page also lets you define a password policy by forcing the minimal length and the use of mixed cases etc.

» Another way to upload files to the server is by using the WebDAV protocol, with which you can access your cloud server from your file manager. If you use Ubuntu, launch the *Files* file manager and press Ctrl+I to enable the location area. Here you can point to your *Nextcloud* server, such as **dav://192.168.3.106/nextcloud/remote.php/webdav**. Once authenticated, the *Nextcloud* storage is mounted and you can interact with it just like a regular folder.

To share uploaded files, go to the Files section in the web interface and click the 'Share' button to the right of the filename. This will bring up a flap where you can specify the users and groups you want to share the file with along with other options such as whether you want to give them permission to modify or further share the file. You can also share with someone who isn't registered with your *Nextcloud* server. Simply toggle the Share link checkbox and *Nextcloud* will display a link to the item that you can share with anybody on the internet. You can also password-protect the link and set an expiration date.

While you can interact with the cloud using the web interface, it's far easier to use one of its official clients. *Nextcloud* has clients for all the major desktop and mobile platforms. These clients also help you synchronise folders from the desktop to your *Nextcloud* server with ease. Many Linux distros, such as Arch Linux and OpenSUSE Tumbleweed include the *Nextcloud Linux* client in their official repos. If your distro doesn't have the *Nextcloud* client in its repos, you can either compile the official client from source or download and use the client for *ownCloud*. The *ownCloud* client is available in the repos of virtually all the popular distros including Ubuntu.

Once the client is installed, it prompts you for your login credentials in order to connect to the *Nextcloud* installation. After establishing the connection, use the client to create a local sync folder under your home directory such as **/home/bodhi/Nextcloud**. Any files you move into this directory will automatically be synced to the server. The client's connection

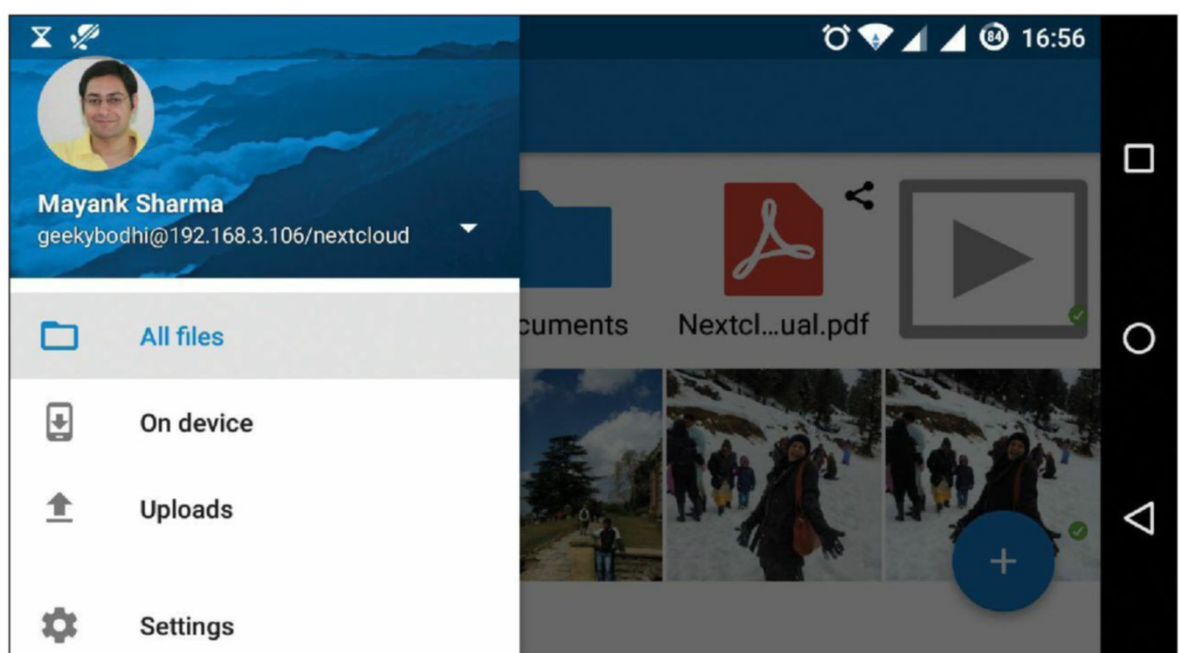
wizard also asks you whether you'd like to sync everything from the connected *Nextcloud* installation or selectively sync files. After running through the client's wizard, you can access it from your desktop's notification area.

When collaborating with other users, you'll appreciate *Nextcloud*'s version control system, which creates backups of files before modifying them. These backups are accessible through the Versions pull-down option corresponding to each file, along with a 'Restore' button to revert to an older version.

In addition to files, you can also sync your calendar and address book with your *Nextcloud* server. Follow the walkthrough to enable the Calendar and Contacts applications. Once you've enabled both programs, the top-left pull-down menu now includes the Calendar and Contacts option. Before proceeding further, you need to import your contacts and calendar from your existing application into your cloud server. *Nextcloud* supports the popular vCard (VCF) file format and almost every popular email application, including online ones such as Gmail enable you to export their address books in this format. Similarly, calendars can be imported in the popular iCal format. Explore your existing mail and calendaring applications and export the VCF and ICAL files for your account before moving on.

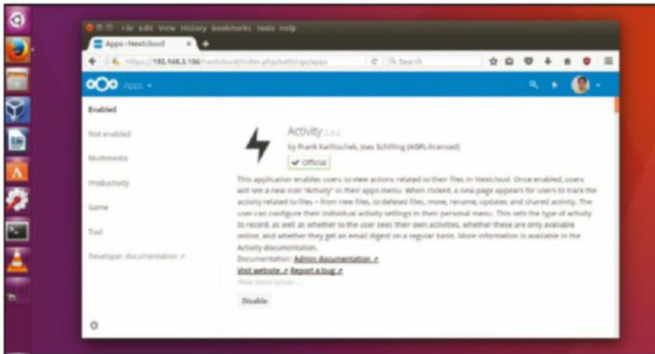
In *Nextcloud* Contacts click on the gears icon in the bottom right-corner of the interface. Select Import from the options that are revealed and point to the export VCF file. The import process might take some time depending on the size of your address book and you can sync these contacts with your desktop and mobile email apps using CardDAV. Similarly, you can import an existing calendar, by clicking on the Gears icon inside the Calendar application. Here again click on the 'Import calendar' button and point to the exported ICAL file.

We've just scratched the surface of what you can do with *Nextcloud*. Follow the walkthrough (see, right) to flesh out the default installation with new applications that will extend the functionality of your personal cloud.



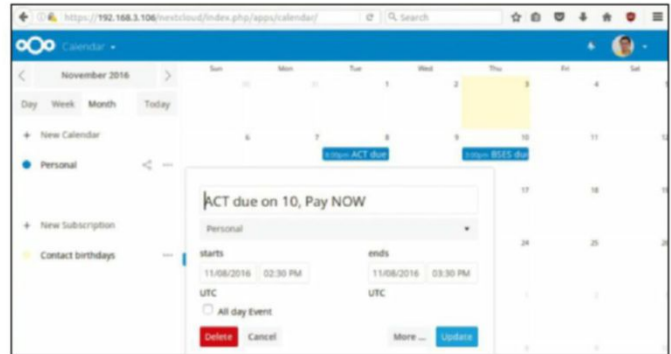
» *Nextcloud* hosts clients for Windows and Mac OS X on its website (<https://nextcloud.com/install/#install-clients>) while mobile clients are best fetched from either Apple's App Store, Google's Play Store or the F-Droid repos.

Get more from Nextcloud...



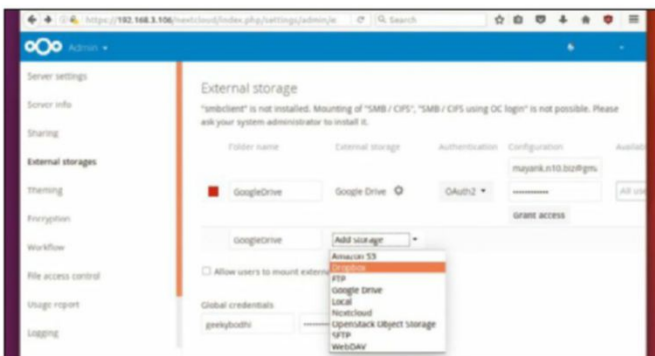
1 Application repository

You can extend your default *Nextcloud* install by adding applications. Bring up the pull-down menu in the top-left of the interface and select the option labelled Apps. By default, you are shown a list that are already enabled. You can browse through this list and read their descriptions to better understand their functionality. You can also disable any enabled app from here.



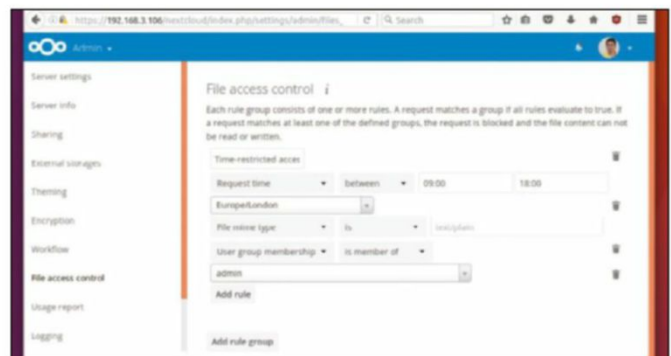
2 Calendar and Contacts

These two should be the first applications you enable. You'll find them listed under the Productivity category. Once enabled, you can use the app's intuitive interface to add dates, contacts and other details. The apps allow you to pull in your existing contacts and calendars, which you can then sync with any PIM applications using industry standard formats and protocols.



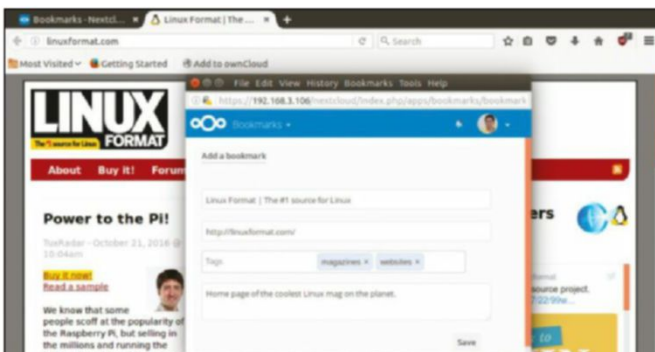
3 External storage

If you still want to use popular public storage services like Dropbox and Google Drive, you can connect and manage them from *Nextcloud* with the External storage support app. Once enabled, the app creates room for itself in the Admin section of the installation. Use the Add storage pull-down menu to select a supported service and enter the authentication details in the space provided.



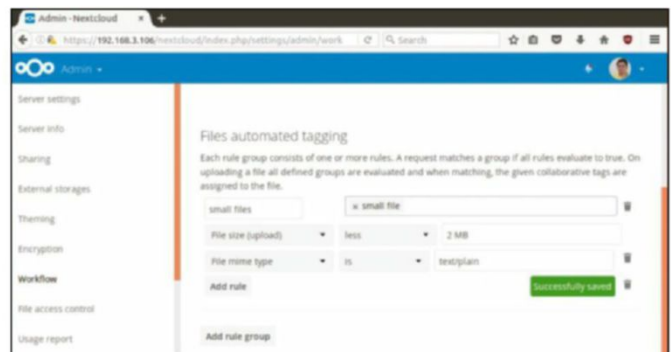
4 File access control

If your *Nextcloud* install will serve several users, it's a good idea to enable the File access control app. This app is also configurable via the Admin section from where you can define various access control rules on parameters such as IP address and file size etc. The rules are tagged to a group of users on the *Nextcloud* deployment and access is only granted only if the attached rules hold true.



5 Bookmark manager

An app that you should enable is Bookmarks. This allows you to store and manage bookmarks in your *Nextcloud* server. Launch the app to store bookmarks directly or import them from a bookmark file from your web browser. The app also has a bookmarklet that you can add to your browser's bookmarks bar. Press the bookmarklet to add a website to *Nextcloud*'s list of bookmarks.



6 Automatically tag files

For better organisation, you can use the Files automated tagging app which will assign tags to files on its own based on conditions as soon as they are uploaded. You can define rule groups for assigning tags in the Workflow section in the Admin section using several different criteria. When a file is uploaded, *Nextcloud* will compare it with the defined rules and tag the file if a matching rule is found. **LXF**

Storage wars

Jonni Bidwell talks flash storage, distributed filesystems and the joys of embracing open source with Sandisk Engineering Fellow, Allen Samuels open source project Ceph.



Allen Samuels is a veteran of the storage industry, having worked for Citrix and Weitek before moving to Sandisk in 2013. (Sandisk and Western Digital merged in

March 2016). He's on the advisory board member for Ceph—the open source distributed object store and filesystem that's set to revolutionise storage as we know it. We caught up with him at LinuxCon in Berlin last December to discuss the technology, the ethos and, as high performance flash-based storage becomes ever more cheaper and reliable, whether or not we should consign all our spinning rust hard drives to the scrap heap.

Linux Format: So why don't you start by telling us a little bit about your career history: where you've come from, how you got to where you are?

Allen Samuels: How much about my history do you want to hear about?

LXF: Let's go back to the very beginning...

AS: I graduated from college and went to work in the mainframe industry, as that's where jobs were available at the time. I worked at Burroughs on their mainframe systems in California. I did that for a couple of years, I left Burroughs and joined Harris which were starting development of a 64-bit super mini computer. I worked on that for about four years and then moved to the Bay Area and joined this fabulous semiconductor startup called WayTek which did maths-intensive chips and graphics. I worked there for about 10 years and then started consulting. I did consulting for about another 10 years and then I started a wide area network optimisation company. I sold it to Citrix, then worked at Citrix for a couple of years, left there and started a cloud storage gateway company for several years. It didn't work out too well. I fiddled around with some other things. And then I joined SanDisk about three years ago. Then we were bought by Western Digital.

LXF: Do you want to talk a little bit about that merger? I mean, there have been a few of these in the storage industry over the last six or seven years.

AS: There's definitely been a lot of consolidation... the architecture of the industry, the ecosystem, who buys what from whom, the products and technology coming from various players has really undergone a lot of change

here in the last few years. I think you're seeing the effects: the converging of old style enterprise to cloud-orientated enterprise. As well as a change of technology you're seeing a company with very different business models, and a different delivery consumption model for customers, and the industry is reorganising itself around that. So a lot of those consolidation acquisitions I think were sort of downstream effects of that...

LXF: Yes, changing markets. So when we think of SanDisk I think of USB sticks, we think of our awesome Sandisk Clip MP3 player, which you can still use even though the clip is broken. Obviously, there's a lot more going on particularly enterprise grade storage and flash-based storage. Is flash storage at a level where it can replace spinning rust?

AS: Well... tape drives never went away. OK? Spinning rust is never going to go away either. I think what you have to realise is that as an engineer I would look at storage as not one thing, but a lot of things. You've got optical discs, you've got tape, you've got spinning rust, you've got flash. We have a set of new entrants in storage-class memory. You've got DRAM, you have SRAM, you have Registers... you put the data where it makes the most sense.

It's really about understanding the difference in performance of those different media. They all have very different physical characteristics.

That's what underlies the choices that you make. So I think when you say "could flash replace hard drives?" I think it's a tiny tale in a much bigger story about how things

are evolving. There's certainly some data that's been traditionally stored on spinning media that will move to flash. But there's a lot more data in the world. A lot of it is most cost effectively stored on hard drives. Today, tomorrow and likely for a very long time. At least as far as we can see right now. If you ask about a particular piece of data, then that answer might change. OK? In general, we've got lots of classes of data that we accumulate that frankly we have to hang on to. But it's very rarely used. At the end of the day spinning magnetic storage is much cheaper on a corporate basis than flash. There's nothing on the roadmaps that suggests that's not going to be true for quite some time.

LXF: SanDisk makes a lot of contributions to



open source, particularly Ceph. Can you tell us a little bit about it for those readers that don't know very much on the subject?

AS: So Ceph is a distributed storage platform that essentially takes a number of computers and a number of storage media and connects them through the network to give you the illusion of a single storage management system. And because it collects a number of them, it's not susceptible to the failures of any individual element. And if one system dies it can be replaced and then reconstructed because of redundant data stored on the other elements in the system. But another very important

ON THE PERSISTENCE OF HDDS

"Tape drives never went away. Spinning rust is never going to go away either."

characteristic is that it's built in a way that as you add more units to the system, the aggregate capability continues to expand.

And when I say capabilities I mean not only the amount of data that's stored, but the ability to transact, to bring data in and out of IOPs (input/output operations per second) or bandwidth. That continues to scale as you add more and more nodes to the system. So that kind of technology is sort of what you probably think about when you think about cloud technology, which is looked at as best practices in the industry.

Ceph is an open source system that does that. It provides object, file and block interfaces, which are the basic standard storage interfaces for applications that are out there, backed by

» cluster storage techniques. It's a project that's over 10 years old. It was started by Sage Weil, I think he did it for a doctorate. I got involved, and got Sandisk involved in Ceph back in 2013 which was when it was really sort of still nascent. And we've been working on it to improve the performance of it, particularly for flash and optimising it. And we've been contributing a lot to that.

LXF: So Sandisk has pretty much been at the forefront of flash-based storage—they and others. But, you know, the first 4GB flash drives seemed amazing. We still have one of those EPCs that had a two gigabyte SSD which somehow still hasn't failed! Is there other corporate involvement in Ceph aside from this question of digital? Are there other players involved?

AS: Oh absolutely. Ceph is used in OpenStack. If you look at user surveys for OpenStack you'll see that the most commonly deployed platform for storage in the OpenStack environment is Ceph. There are a number of other corporate contributors, such as Intel which contributes to Ceph, SUSE contributes to Ceph. There's a fair number, I think there are big contributions from Seagate and Samsung. It's sort of the usual suspects in the storage world which contribute.

LXF: It's great to see companies embracing open source, rather than – I guess how it

was at the beginning – turning their nose up at it and then maybe being a little bit scared of it later on. It just makes sense, I think.

AS: You know I think the vendor relationship with open source is what changed. It's really the vendors that react to what the customers want. There's a sort of sea change in the relationship of the industry to open source, which I think reflects the quality and the energy that's been put in by the open source community. They've certainly evolved from the early days, and recognized that in order to be successful they had to take on some of the characteristics that businesses want – to have something stable that can be serviced and has good quality documentation. You know, the things that make a product usable by customers. You see increasing attention to that in the open source community. So I think it's the community and the vendors reacting more to customers, rather than the other way around.

LXF: It's good when companies listen to the people. So I mean ... the picture I have of Ceph now is this, sort of, a meta filesystem almost, which governs files on other things. Is there going to be an actual filesystem underneath that? And we've got things like F2FS and Btrfs coming up now... well Btrfs is meant to be friendly to flash storage. Are there any recommendations or things you shouldn't use underlying a cluster?

AS: Well, the current best practices for Ceph is to use the XFS filesystem. Historically, there was support for Ext4 and Btrfs. In the latest release, the Ext4 support has been deprecated. Btrfs has not seen production in Ceph. The community are, in fact, deeply involved in a redesign of the lowest level interfaces of Ceph, which will actually remove it from being on top of the filesystem. So instead of utilising a native filesystem, it's going to use a storage block itself and manage that itself. And I can certainly get into the reasons for that if you are interested. It's an instance of what you see generically in the storage world. Which is the storage problem, it's what I call 'short sheet', which is like trying to fit a short sheet on a bed. There's no one size fits all solution. If you tweak something then you yank the sheet in one direction, and you're going to trade off against something else and make that worse. So if your head gets warm, your feet get cold or vice versa.

In the case of Ceph and filesystems, you had filesystems with a set of semantics that are frankly ancient. One of the things that Ceph provides is what's called strict consistency across the nodes, so when you get outside of an individual computer and you start to think about arrays or groupings of them, the issue of simultaneity becomes a real problem. It turns out that actually Einstein was right! Now we know that when I talk about how our filesystems are Newtonian in the sense that



there is a God's eye view of the world. Everything is ordered and you're either ahead of me or behind me. Either I go first or you go first. Clusters of computers are Einsteinian, OK? And there is no absolute time. Things are path dependent and the underlying semantics you use to bridge that gap between the two is something that Ceph does for you. So Ceph provides what's called strict consistency, which would be the Newtonian view of things. But in order to do that, it wants to have a storage system with a certain set of characteristics between... transactional characteristics that a filesystem simply was never intended to supply.

So what you have in the Ceph world is an infrastructure built around transactions on a foundation which doesn't understand transactions! And somehow make that work. It turns out it consumes a lot of resources. We're in the process of redoing that layer and creating a lower layer that natively understands transactions and doesn't have this sort of mismatch between the two, and will perform much better as a result.

LXF: We can imagine people 20 years thinking about this stuff and thinking "Yeah, it would be very nice to store a little bit over here and a little bit over there". Obviously you run into all kinds of complicated problems and that's why it's taken a lot of bright people, such as yourself, to get to this stage. Can you talk in layman's terms about some of the problems you come up against? I mean I think latency is the first one that comes to mind... especially when you're dealing with redundancy. How do you make sure everything gets to all the places that it's supposed to go at roughly the same time?

AS: Well, you have the change that's going on in

ON MANAGING SCALE

"When you get to 100,000 [processors] it's a completely different problem."

the storage industry... it's really the tail end of Moore's Law at the end of the day. Things are continuing to get cheaper, on the electronics side. And you get more transistors for the same dollar every year. Your question is what are you going to do with that? It's been a long time since we had single core CPUs. So, you know, now all the CPUs have multiple cores in them. It turns out there's only so many transistors you can consume doing one thing and making it faster. After a certain point you get into diminishing returns. It takes more transistors to do things a little faster. So now you want to do

two things at a time and so on. If you scale that up, you now have armies of computers – literally hundreds of thousands of servers – and at that scale the problem becomes completely different. How do you deal with that?

We're just now on the verge... and you see this primarily in the data world... with building abstractions, paradigms, templates, of how to efficiently utilise this much computing at this scale. When I started in the industry quite some time ago, you had one computer. That was all you needed to work with. And then suddenly it was two processors and that

was a big deal. But three or four, those are hard. But then when you get to 100,000 it's a completely different problem.

That, I think, is one of the biggest challenges for the industry right now: learning how to process data at these kind of scales. When I commute to work every day, I usually pull my phone out to check what's the best way to get to the office, avoiding routes if the traffic is pretty heavy. People say 'this is great! You know we could never have had this 20 or 30 years ago!' Actually, we could have. There's nothing in the technology that didn't exist 20 or 30 years

ago, it just cost five orders of magnitude more. The actual value of that to me is not worth a lot. OK, if I had to pay a million dollars for my phone and its various apps I wouldn't pull it out and check it every morning. But because for all intents and purposes that's free to me, I do that all the time. So as the costs continue to come down, the scale goes up, all sorts of things that were prohibited before become possible. And who knows where it's going to end up.

LXF: You have previously mentioned that Ext4 support is being deprecated. Do you think that it's just a sort of symptom of looking to get rid of the underlying filesystem altogether. Or are there particular deficiencies with Ext4 that the project wanted to avoid?

AS: We found that there were elements in some very obscure corners of the Ext4 filesystem that just didn't behave the way they were expected to and frankly there are very few people that were using it. Rather than trying to come up with some fixes that would be horrendous and make maintaining backwards compatibility very difficult, we decided to deprecate support. The belief is that there aren't any projects relying on Ext4, so that's basically why. The move away from filesystems in general is sort of unrelated to that. **LXF**



Invisible Internet Project

Roll Over Tor, Nate Drake explores the strengths of the shadowy I2P network.

Quick tip

For an in depth comparison of I2P vs Tor visit <https://geti2p.net/en/comparison/tor>

T I2P (The Invisible Internet Project) is a form of 'darknet' allowing users to visit web pages, chat, blog and even BitTorrent securely. A more detailed technical rundown is forthcoming but for now it's enough to know that this anonymous peer to peer network provides the strongest protection against attempts to unmask your location and/or identity. This is mainly because it has been built from the bottom up with privacy in mind, but it also is largely unknown in privacy circles when compared to more popular alternatives such as Freenet or Tor. We are sure you can appreciate the irony.

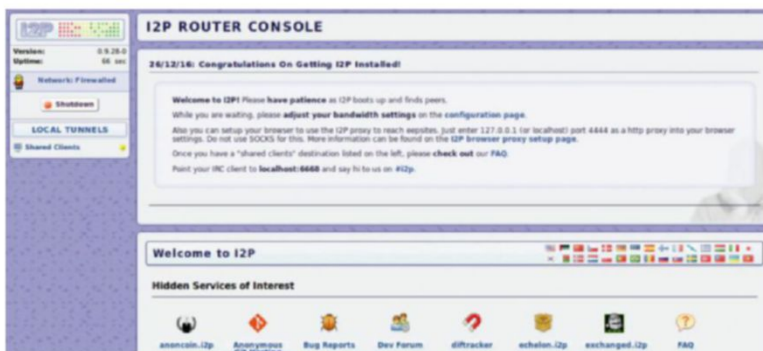
Readers interested in their own security may well already be aware of Tor and its trusty companion the Tor browser bundle. (See www.torproject.org if not). Tor is also designed to anonymise your connection on the internet. The method used to achieve anonymity through I2P is similar to Tor's. In fact, one of the safest ways to access I2P's own 'eepsites' is to use the Tor Browser Bundle or the custom *I2P browser* in the TAILS Operating System.

As such, is I2P trying to reinvent the wheel, given that we already have Tor? In order to answer this question, it's essential to understand Tor's weaknesses. Tor uses 'onion routing'—a process for sending data anonymously over networks. Packets of data are encrypted and sent through multiple nodes run by volunteers throughout the world.

Each Tor relay only knows the previous location of an encrypted packet as well as the next node to which it is being sent. It's like peeling an onion; individual layers only touch those immediately in front and behind them. All the others remain totally separate to the others.

Paths for data through the Tor network (known as 'virtual circuits') are chosen randomly and when data packets reach their destination they can be decrypted to display information such as a web page. Each new request involves another random path of encrypted data being passed through multiple nodes making your location extremely hard to trace.

The Tor project is quite candid: onion routing isn't perfect. By default, for instance, Tor makes no attempt to conceal the



➤ The I2P Router Console will launch in your default browser once I2P has been installed. Scroll down to see some useful hidden services and applications.

fact that you are accessing its network so an adversary may not be aware what you're accessing but will know you are attempting to hide something. Tor also cannot protect data at the boundaries of its network. In other words, if an adversary can monitor data both entering and exiting the network it can perform 'end to end correlation' of traffic

This was readily demonstrated by Harvard student, and all-round silly sausage, Eldo Kim in December 2013 who used Tor and a disposable email address to send fake bomb threats in an attempt to avoid his final exams. Kim blundered however by using the University's own Wi-Fi network, so it was a simple matter of checking who was using Tor on campus at the time—the lone Tor user was Eldo. Even those not silly enough to avoid using the same wireless network as a recipient cannot always protect themselves fully. A shadowy government organisation, for instance, who can see traffic both entering and leaving the Tor network across state borders may still be able to trace your location

People accessing the 'clearnet' (regular internet) through Tor are forced to rely on 'exit nodes' to relay content. Tor cannot, by design, protect your traffic against this, so if you access information on a page not protected by SSL, a malicious person running a Tor exit node may not only be able to discern your location but glean details about sites you visit, emails you write and even potentially passwords

Finally, filesharing via BitTorrent over Tor is discouraged as it can potentially leak your real IP address and places an undue strain on the network.

Tao of I2P

I2P is preconfigured to provide a greater level of privacy out of the box. In the first place, every I2P peer participates in routing data to others. Every ten minutes a connection is established between an I2P user's machine and a peer. There are no entry and exit points to compromise. This makes it much harder to match any transfer of data over the network to you specifically. Messages are also encrypted and bundled together making it almost impossible to distinguish which belongs to you in particular. This is known as 'garlic routing'.

Instead of establishing virtual two way 'circuits' to transfer data as with tor, I2P uses unidirectional 'tunnels' to either send or receive data. This doubles the number of nodes that

"I2P uses unidirectional 'tunnels' to either send or receive data."

an adversary would need to compromise before they even have a hope of accessing your information, as well as making it harder to determine the type of data requested e.g. a web page. Unlike Tor's virtual circuits, I2P's tunnels are short-lived making them harder to trace.

I2P is also designed specifically to make use of hidden services. Accessing eepsites is generally much faster than navigating through Tor's .onion domains. This is put to good use by I2P's built in Bittorrent client *I2PSnark* (see *Torrenting With I2PSnark, over the page*),

which uses trackers provided by its own network to allow you to download files anonymously.

I2P benefits from 'security by obscurity' due to its smaller user base it hasn't been targeted as

much [that's what they want you to think!—Ed] by law enforcement, as was the case in 2013 when the FBI allegedly placed malicious JavaScript in certain pages running Tor hidden services to reveal the public IP address of users.

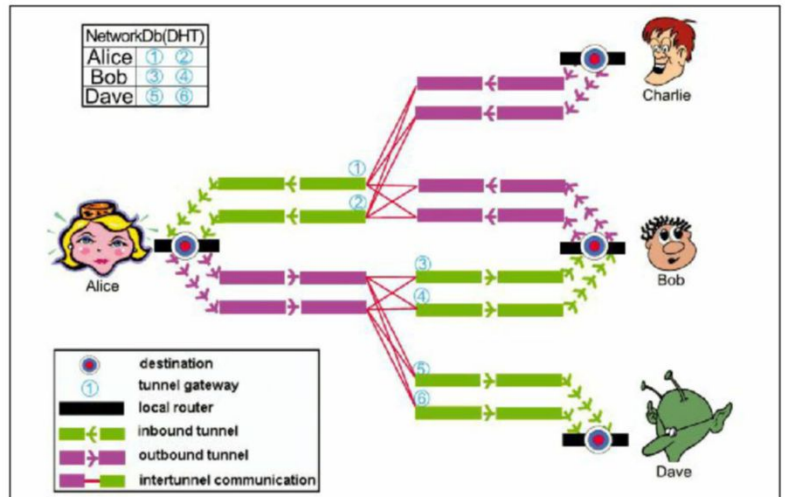
Choose your browser

Having sung I2P's praises, hopefully by now you are eager to plunge in and download the Java installer. The I2P program itself acts a local proxy, the idea being that you can configure your web browser or similar to channel traffic through it.

» **Garlic routing at its finest. I2P uses separate 'tunnels' for sending and receiving information.**



Quick tip
Eepsites are I2P's own hidden addresses. They use a pseudo top-level domain '.i2p'. For instance, the I2P's own project page is <http://i2p-projekt.i2p>.



Garlic & Onion

Head to www.torproject.org/download/download and download the *Tor browser* for Linux. Extract it using your favourite utility. When the Tor browser runs you'll be asked to configure your connection. Simply say that you connect directly to the internet and launch the *Tor browser* for now.

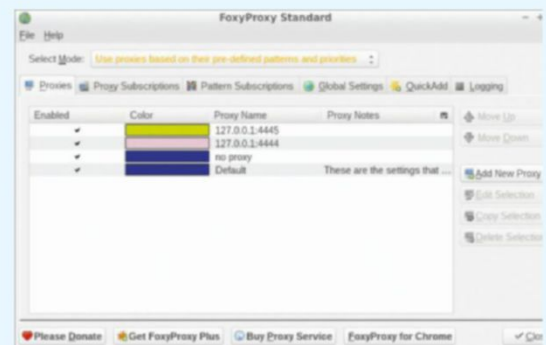
Next click Tools > Add Ons. Click 'Add ons' on the left-hand side of the Window and enter 'FoxyProxy' in the search bar at the top right. After a few moments the options will load. Click the 'Install' button next to FoxyProxy Standard. The Tor browser will restart.

Next, download the configuration file for *FoxyProxy* from the Tin Hat website: <https://thetinhathat.com/tutorials/>

darknets/foxyproxy.xml. Enter the address in the browser and save the XML file to your computer.

Go to Tools > Add Ons once again and hover your mouse over FoxyProxy Standard. Click 'Options' to open the window. Click File > Import Settings. A pop-up window will open. Choose the .xml file you downloaded. *FoxyProxy* will restart the *Tor browser*.

You can now use the *Tor browser* to access both Tor hidden services (.onion addresses) and eepsites (.i2p addresses). Once you're sure the Tor Browser bundle is working correctly copy the entire **tor browser** folder to a secure medium such as an encrypted USB stick for extra security.



Import the .xml file to load this configuration into FoxyProxy. This will make sure that you can access eepsites without interfering with the Tor Browser's other functions.

» The software running I2P is called the *I2P Router* and when following the instructions in our guide (see right), by default it will open up the router console's main page in your default browser. The router software will need some time to establish connections to enough peers to work well.

Once I2P is running you will need to configure a web browser to use the local proxy in order to be able to access eepsites with the extension .i2p. Note that even if you do this, normal 'clearnet' websites such as **www.linuxformat.com** will continue to load in the same way.

This is excellent in that you can continue to use your normal web browser as before, but bear in mind that when visiting regular websites your location and web traffic will be as visible as ever. This means you could be identified through the process of 'browser fingerprinting'. This form of surveillance analyses browser settings such as screen size, language and plugins to build a picture about you. There's also the risk when visiting clearnet websites of accidentally installing malware.

In light of this, try to install a separate dedicated browser to access .i2p sites. The Tor Browser bundle while primarily designed to help you access the Tor network, can be used for this purpose. Better yet, it's based on a long term support version of the *Mozilla Firefox* browser with a selection of plugins which enhance privacy, reducing the risk of browser fingerprinting. You can even still access tor hidden services as well as I2P's if you wish. (See *Garlic and Onion box*, previous page, for steps on how to set this up.) If you are ultra-paranoid (we hope you are), consider downloading the privacy

conscious operating system Tails: **<https://tails.boum.org>**. Tails – which stands for The Amnesiac Incognito Live System – can be run directly from a DVD or USB stick leaving no trace on your computer after use.

Tails 2.5 (covered in Features, p34, **LXF216**) has a dedicated I2P browser, saving you the trouble of downloading and setting up your own. Simply hit Tab when booting, hit the Spacebar and add the word **i2p** to your boot parameters.

I2P sites can only be accessed via the Tails *I2P browser* while all other traffic is channelled through the Tor network and the browser is run from a dedicated user account, further increasing your anonymity.

I2Peeved

I2P can only promise anonymity within its own darknet. If you choose to check your Gmail inbox online at the same point you're accessing an eepsite, although it may not be possible to know what you were doing on I2P an adversary may be able to prove you were online. Read the Choose your browser section carefully and pick accordingly. If you must use your regular web browser to access I2P, consider opening pages in a private window which won't record your history.

The tutorial for installing I2P assumes you have an implementation of Java on your machine, such as 'OpenJDK'. The OpenJDK website has installation instructions for all major versions of Linux at **<http://openjdk.java.net/install>**. Ubuntu, for instance, needs to enable the universe and multiverse repositories.

If you're using a Debian-based distro such as Ubuntu, Linux Mint or Knoppix, you can also install I2P without using Java at all by adding the relevant repositories. See **<https://geti2p.net/en/download/debian>** for more information.

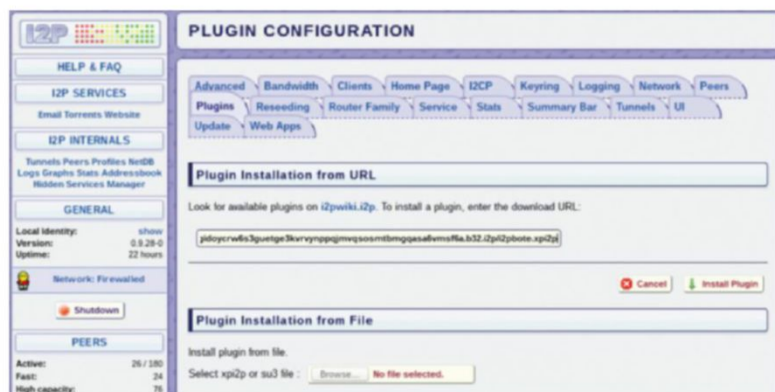
As eepsites have to be maintained by active nodes, they can quickly fall out of use. Visit **<http://identiguy.i2p>** for a list of active .i2p sites.

While I2P speeds compare favourably with Tor, the default bandwidth of 32Kbps will not result in very zippy browsing. Consider increasing your bandwidth from the router configuration page at **<http://localhost:7657/config.jsp>**.

I2P like Tor can be vulnerable to 'intersection' attacks where an adversary periodically analyses the nodes which are online while you are using the network. The resources required for this are immense given that I2P tunnels that are used to transfer data vanish quickly. For a full rundown of this and other threats which might reveal your identity and/or browsing habits visit **<https://geti2p.net/en/docs/how/>**

Quick tip

See **https://tails.boum.org/doc/anonymous_internet/i2p/index.en.html** for more information on using the I2P Browser from within TAILS.



» Install plugins such as I2PBote via a URL or downloaded files. They will appear in the Applications and Configuration section of the I2P Router Console.

Torrenting with I2PSnark

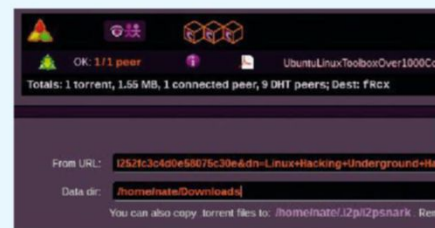
Head to the I2P Router's home page (**<http://127.0.0.1:7657/home>**) and scroll down to Applications and Configuration and click 'Torrents' to open *I2PSnark*. Once the application opens, ignore the main options for now and click the 'Configure' button at the very bottom of the screen. The Data Directory field shows the location where files will be downloaded. By default this is **~/i2p/i2psnark**. Change this to another folder or external drive if you wish.

You can change the dark default ubergine theme to something a little easier on the eye here if you wish, as well as change the number of tunnels to be used for incoming and outgoing BitTorrent connections.

Scroll to the bottom of the page to examine the trackers. As you will note these are all eepsites with .i2p domains. This means you can download any files you find anonymously via I2P's darknet but you will be unable to download files via BitTorrent from clearnet sites such as Pirate Bay—please respect copyright laws.

Click the 'Save Configuration' button then click on 'I2P Snark' at the very top of the page. At the very top of the page you'll find links to the main Torrent eepsites on I2P, Diffracker (**<http://diffracker.i2p>**) and Postman (**<http://tracker2.postman.i2p>**). The sites are very easy to search, once you have found your file of choice simply right click on the magnet symbol to copy its link. On the I2PSnark page, paste this link into the

From URL field then click on the 'Add Torrent' button. The file will be downloaded to the data directory you specified earlier.



» Paste the magnet link into the 'From URL' field. You can also change the Data Directory if you wish before clicking 'Add Torrent'.

threat-model. Although this tutorial has focused on web browsing via I2P, the network is capable of much more such as email and IRC chat through use of the application I2PTunnel. See <https://geti2p.net/en/docs/api/i2ptunnel> for more.

Enhancements can be made to the I2P Router software through adding plugins via the Configuration page (<http://127.0.0.1:7657/configplugins>). One extraordinarily useful application is I2PBote (<http://i2pbote.i2p>), which

provides an end-to-end, anonymous email system, capable of sending messages to regular email addresses as well as those ending in .i2p.

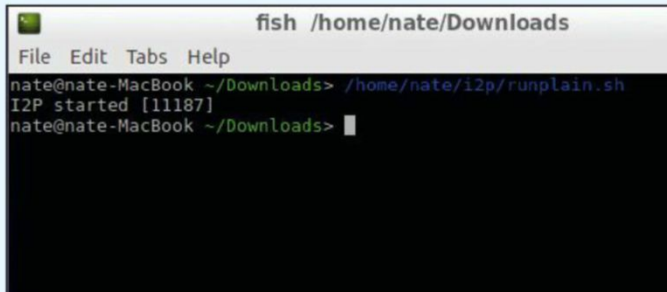
I2P is still technically beta software and hasn't been subjected to intensive peer review. As ever, take a compartmentalised approach to your privacy. If you have suggestions or feedback about your experience with I2P, head over to <https://geti2p.net/en/get-involved> for details of the team's IRC channel and their eepsites. **LXF**

Install I2P



1 Download and launch Installer

Head to <https://geti2p.net/en/download#unix> and click the **i2pinstall_0.9.28.jar** button to download the I2P Java installer. Save the file to your **/Downloads** folder then open the Terminal and run `java -jar i2pinstall_0.9.28.jar` to launch the installer. You may need to change the filename and you'll need Java.



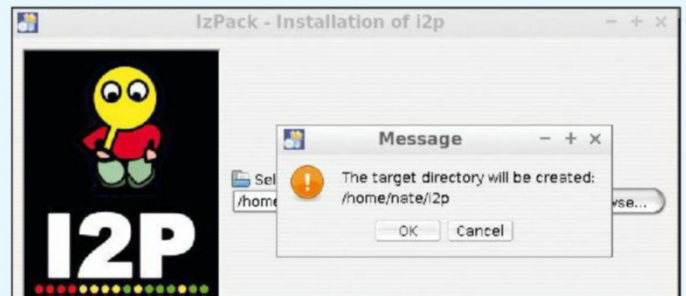
3 Launch I2P router

Open the terminal and use `cd` to navigate to the directory where I2P is installed e.g. `cd /home/alice/i2p`. Next run the script to launch the I2P router with `./runplain.sh`. You will see a message indicating I2P has started. The I2P router console will open in your default browser. Give I2P a few moments to establish connections.



5 Configure the proxy

I2P acts as a local proxy for web browsers to allow them to access eepsites. Specific steps for individual applications are available on I2P's website <https://geti2p.net/en/about/browser-config>. If you are using the Tor Browser bundle, configuration is done automatically through the FoxyProxy add on so you can ignore this step.



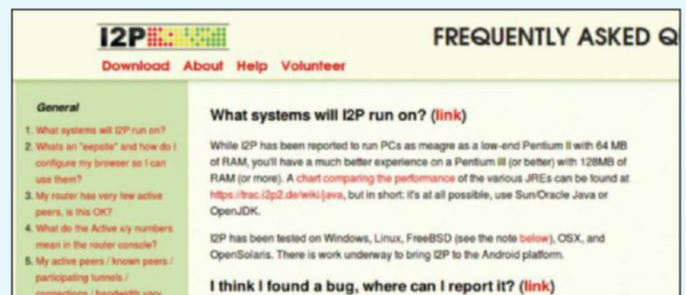
2 Setup I2P

Select your language of choice, then click 'Next'. Read through the details of I2P being in the public domain and click 'Next' again. You will now be asked to specify an installation path for I2P. By default this is `~/i2p`. Once installation is complete the installer will list two commands that can be used to launch the I2P router itself.



4 Download a browser

This step is optional but recommended. Although you can technically configure your usual web browser to be able to navigate to .i2p (eepsites) this isn't very secure. Consider downloading a separate browser at this stage to work specifically with I2P. For extra security consider accessing eepsites using the *Tor browser*.



6 Test your I2P browser

Restart your web browser and try to navigate to <http://i2p-projekt.i2p/faq>. If you are successful, not only will you know you can access eepsites but you can read through some common questions about I2P. Before you begin browsing, navigate to 'Why is I2P so slow' for instructions on increasing your bandwidth.



Jolyon Brown

When not consulting on Linux/DevOps, Jolyon spends his time bootstrapping a startup. His biggest ambition is to find a reason to use Emacs.

Mr Brown's Administeria

Esoteric system administration goodness from the impenetrable bowels of the server room.

Volunteers needed

A small news story that caught my eye recently was the reported issues with the NTP Pool project (www.pool.ntp.org). Traffic to the pool increased by a large amount (something like a twentyfold increase at its highest point). After some investigation, it was determined that a buggy version of the Snapchat app on the iPhone was to blame and the company to its credit dealt with it very quickly. But it did highlight that this widely used resource – timeservers used by millions of clients – runs on systems donated by volunteers.

At the time of writing, there are 3,606 servers in the worldwide pool which seems quite a small amount in relation to the amount of work they actually do. Of course, it's likely in your professional capacity that your systems are synchronised to a paid-for service.

Spare some time

This is absolutely the thing to do when having your systems set to the correct time is vitally important. But it's also probably the case that more systems that you think (especially at home) are quietly using the pool, which is only going to get busier and busier. So my appeal to readers with a static IP address and a permanent internet connection (and a bit of spare time) is to donate a few hundred bytes per second to the project. On the project website (take a look here: www.pool.ntp.org/en/join.html) are instructions for how to join and set up a server. I'm sure the type of people who read this column will have no problems getting one up and running (and if you haven't done this kind of thing before it's a fun little exercise). This is a long-term commitment (read the caveats on the project site!) but I think it would be awesome to hear from LXF readers who contribute to something which helps the entire internet community. jolyon.brown@gmail.com.



OpenAI opens a universe of learning

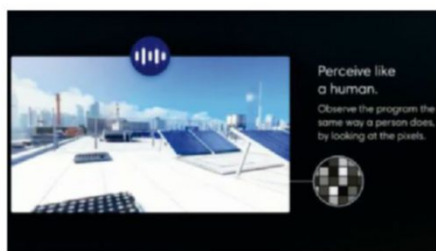
Teach your computer to play games then sit back and watch it try to beat your high score.

OpenAI, the Silicon Valley non-profit dedicated to building 'safe' AI (and which has backing from several well-known billionaires) recently released Universe (<https://universe.openai.com>), labelled as 'a software platform for measuring and training an AI's general intelligence across the world's supply of games, websites and other applications.' The software allows an AI agent to use a computer in a similar manner to a human (via keyboard, mouse and screen). Using VNC, Universe presents a program to an agent via a remote desktop. The agents get no special access to any program internals or APIs etc. The project website announced that many well-known games development studios have granted free access to agents using Universe to play their games and it's hoped that many more will follow suit.

Of course, games are not the only programs that could be accessed via this route, but they are a good model for testing reinforcement learning type agents (they have a defined, measurable reward for example, such as gaining a high score). OpenAI also provide a project called Gym which is a toolkit for these kinds of agents.

The goal of the project is to help develop a single AI agent that can use its experience in Universe-provided environments to quickly master unfamiliar ones. This would be a step towards the goal of a machine demonstrating general intelligence. This type of all round, problem-solving ability remains out of reach for systems at the moment although there are many ongoing efforts in this area. Universe is available to download from GitHub (<https://github.com/openai/universe>).

The games themselves generally run in Docker containers and Universe is largely written in Python. Around one thousand Flash-based games are available with Universe, as well as several from Atari (from the 2600 era) and a lot of example code for writing agents etc. The OpenAI team would like contributions from the community and having had an initial play around with it, the project is tremendous fun and will hopefully feature in a future episode of the magazine (assuming agents haven't replaced us all by then).



› I don't get time to play games as it is, so I may as well let my computer play them for me.

Part 1: DC/OS

The 'Datacentre Operating System' is a contender for running containers as well as big data applications.

Another issue, another column on orchestrating containers? I can hear the exasperation from the **LXF** readership already. But wait, before you flip the page and go looking for Hot FOSS (Hint: see p48) and whatnot stay just a little while (i.e. don't go to p48) and indulge me while I talk about DC/OS, one of the bigger contenders in this crowded market. DC/OS is relatively new as an open source project (having only been released in April of 2016) but has its roots in the Apache Mesos project (which in turn came out of the Berkeley as an open source version of Google's Borg system) and is backed by a company called Mesosphere (the original authors, who sell an enterprise version). Anyone who used Twitter a few years ago might remember the company being almost as famous for the regular appearance of the 'fail whale' as it was for being the hottest social media company of the day. Mesos was apparently the technology that finally cracked Twitter's scaling issues. Not that Twitter doesn't have plenty of other issues left to deal with of course, but I digress.

DC/OS positions itself as being able to handle really large – data centre-sized in fact – workloads. It feels a lot more heavyweight than the likes of Rancher that we've covered here in the past although many of the principles are the same. A big mark of faith for just how heavyweight and robust it is came from Microsoft, who has chosen elements of it to run the Azure container service (part of its AWS rival).

Pooling of resources

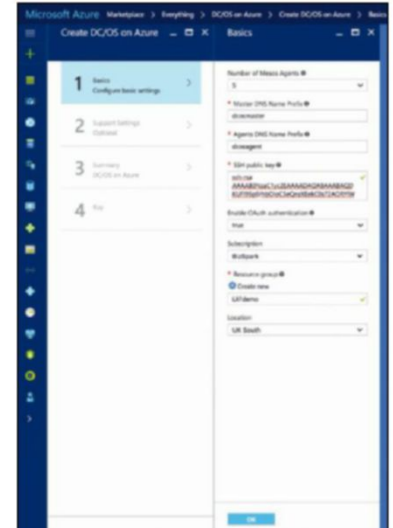
DC/OS is actually a bundling together of several open source projects into a cohesive whole. As well as Mesos itself, these include Zookeeper for service discovery, Marathon (written by Mesosphere originally) to provide orchestration and Exhibitor (used to install and configure Zookeeper, originally written by Netflix). The project believes that this bundling and associated polish mean that the software is greater than the sum of its parts. However, Mesos is undoubtedly the core on which everything else stands and is worth considering on its own for just a moment.

Classing itself as a distributed kernel, Mesos actually consists of a few elements, the first of which are a master daemon which manages agent daemons. There can be multiple masters, which discover each other via Zookeeper and elect one of their number as the leader. The remaining masters act as standby systems in case the leader becomes

unavailable. All pretty standard stuff. Agents look to try and connect to the leader (via the Zookeeper discovery service) and register to it. The agents report back to the master about their available resources (number of available CPUs, amount of memory and such like). As well as the master/agents, Mesos has the concept of a framework that runs tasks using the reported resources. This needs a scheduler to examine resource requests and an executor to actually start tasks on nodes. Basically, the scheduler gets offered reports of resources from the master which it ignores until an end user triggers a job of some kind on the system. At that point, it compares available resources to the job requirements and asks the master to handle sending the tasks to the agents which, in turn, use the executor to actually start them. As the task completes, its status is reported back up this chain back to the initiating client. Don't worry, it's a lot simpler than it sounds!

Mesos (and by extension, DC/OS) states that this separation of resource management from task scheduling means that multiple workloads can be collocated efficiently, getting the most out of compute resource. Indeed, Apache claim that Mesosphere can scale (and has done) to tens of thousands of nodes. While the average **LXF** reader won't have that many machines lying around (although my garage could probably pass for a data centre given the number of old computers in it) most cloud providers do—and it's here that DC/OS aims to make its mark.

As well as being scalable, mixed workloads – containers, web applications as well as big batch jobs and analytic type tasks – can be run on the same hardware. DC/OS uses Marathon to schedule some of these, but other schedulers (e.g. Hadoop) can also run alongside it. This is quite a difference from container only platforms. That's not to say DC/OS is a slouch on this front. Marathon can handle container orchestration quite happily and comes with all kinds of grouping and placement rules which can be applied (more on this next month).



» Azure is pretty slick when it comes to standing up infrastructure although I cursed when an error appeared a couple of seconds after this screenshot.

»

Differences between DC/OS and Kubernetes

If I had to pick a market leader in the container management space at the moment, *Kubernetes* seems to be the project to look at (although *Docker* is constantly adding to its open platform). The Google-backed open source version of its own in-house system has grown in popularity in the last year and there seems to be a scramble to support it within other offerings (e.g. *Rancher* and *OpenStack*).

So how does it compare to DC/OS? *Kubernetes* doesn't attempt to handle the same range of applications that DC/OS and Mesos do, being purely container based. Mesos is a bit lower level than *Kubernetes*, concerning itself with CPU and memory resources (the whole 'distributed kernel' concept) and DC/OS covers a wider range of target applications (as we shall see next month). *Kubernetes* can actually run

on Mesos as it happens—a project to do this is available via GitHub. So, which is the better option of the two? I must admit to liking both!

Competition is good and drives forward innovation. If I was purely working with containers then job market demand might drive me to *Kubernetes*, but DC/OS is definitely worth knowing about and having a go at, as I'm hoping to demonstrate in **LXF222**.

» What else does DC/OS provide besides running jobs across multiple machines? Well, there are some options on storage with ephemeral and persistent volumes available (generally locally based). There are a bunch of packages that can be installed via the fancy titled 'Mesosphere Universe Package Repository'. This is a common feature set for this kind of software, though. There are a number of ways to get up and running with DC/OS itself, such as public cloud options, local hardware and/or virtual machine setups and hooks into config management systems.

Scaling can be done in a manner similar to other platforms. Adding extra nodes to DC/OS is a piece of cake, while horizontal and vertical scaling is handled via Marathon (in some cases automatically in response to load). The whole platform has high availability designed into it (so long as multiple masters are used) and the services running on it are monitored by Marathon and restarted should they fail. Upgrades to DC/OS are 'zero downtime' with multiple deployment options available (such as rolling, blue/green and canary type scenarios). Finally, in addition to Zookeeper based service discovery, load balancing elements are built into DC/OS with Layer 4 (transport) and Layer 7 (application) choices available, automatic DNS endpoint generation, a handle CLI and a well-designed web front-end to show off to your boss.

To the clouds

Well, I'm sure you're convinced by now that DC/OS is the answer to your dreams and can replace whatever hoary old stack you spend your days supporting at the moment. But don't start destroying things just yet. Let's get an example stack stood up and ready to play with ahead of next month. For ease of use, I decided to use Azure this time around

(other cloud providers are available) because I had some credit with them as a result of a subscription that I needed to use up! The things I do for the readers of **LXF**...

As it happens it was pretty straightforward to set up DC/OS in this manner. Azure has a marketplace where it can be selected. After that, it was a couple of wizard type screens, similar to most other cloud providers if I'm honest. The DC/OS install guide suggested choosing five agents in order to be able to use all the services it had available, and I had to supply an SSH public key (needed to access the system afterward). The whole operation took around eight minutes (at the second attempt—I tried to stand it up in the newer UK South location at first only to get an error that a particular machine type was unavailable at that location). But then I was presented with a variety of objects in my dashboard—virtual machines, load balancers, a new public IP address and some network security groups and interfaces.

From there I had to retrieve the value of 'MASTERFQDN' from the output of the deployment. This was the name of the system where I had to tunnel in order to finally see the DC/OS dashboard (and was a typically horrible cloud system type name: **dcosmasterwqhnhwxtdytst.westeurope.cloudapp.azure.com** or some such). Having had enough Microsoft branded screens for one day, I could now switch to good old fashioned SSH:

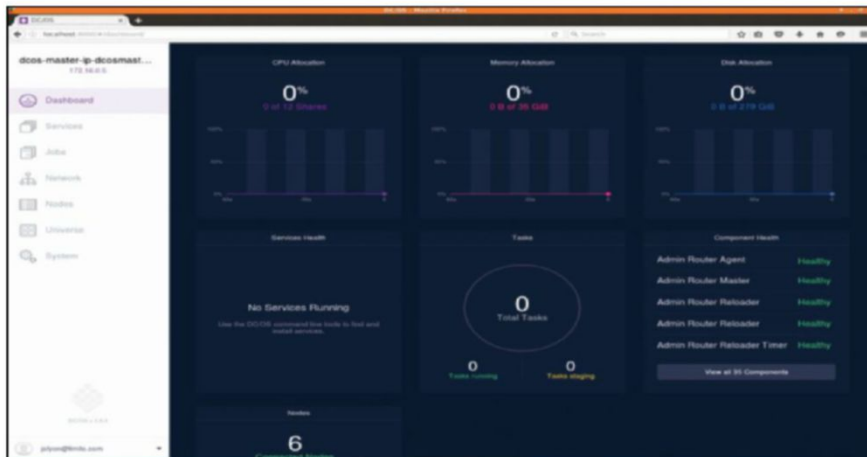
```
$ ssh azureuser@dcosmasterwqhnhwxtdytst.westeurope.cloudapp.azure.com -p 2200 -L 8000:localhost:80
```

Firing up my browser and connecting to port 8000 on localhost brought up a nice login screen. Since I chose to enable OAuth authentication I was asked for a Google, GitHub or Microsoft account to use at this point. Entering my details (and two-factor details) I was presented with the initial dashboard screen which can be seen on this page. A brand

new shiny DCOS installation ready for use! I took a quick flick through the screens which showed me my nodes, details of my newly formed cluster, a handful of packages I could install (such as Jenkins, GitLab, Cassandra and Spark) and the individual components available for me to try out.

Next month I'll be doing just that—putting DC/OS through its paces to see how it compares to some of the other systems we've looked at over the last year.

» A nice clean install of DC/OS ready to use. Five nodes provisioned in just a few minutes.



I want to pay for support!

As mentioned in the main article, Mesosphere (<http://mesosphere.com>), one of the authors of Mesos, provide commercial support for DC/OS with their Mesosphere Enterprise DC/OS product. But what are the differences between that and its open source sibling? As might be expected, support options play a big part here.

Contracts are available for 24x7 and 9-5 offerings, with prices based on the number of nodes (physical or virtual) that exist within the environment. It looks as though the enterprise

version also adds a bit of extra polish that goes down particularly well in the corporate environment; LDAP/Active Directory integration (for identity management), secrets management (for secure storage of credentials and keys) and role based access to containers, jobs and services are all sure fire ticks in the box for the average corporate purchaser of software.

As well as this, the company bundles in some 'enterprise monitoring and troubleshooting tools' although there isn't a huge amount of

detail of how this differs from the open source equivalent. I did find some mention of enhanced logging which would please a corporate security team. There's also a brief mention of multi-tenancy which is often a must-have for service providers. This is where clients, for example, might share infrastructure they rent but thanks to access control, they are unable to see each others applications or data. For anyone running say, some kind of data analysis as a service for financial clients this is worth paying for by itself.

HAProxy 1.7

A brand new version of HAProxy gives **Jolyon Brown** an excuse to revisit one of his favourite open source projects.

With the *HAProxy* project recently releasing version 1.7 of its software, I wanted to take a little bit of space up in this month's Administeria to shed a little bit of light on this milestone and to remind people how useful this package is. Readers might be bored with my anecdotes of how open source projects like this have liberated infrastructure development, but back in the day the kind of functionality given by *HAProxy* would only have come in the form of physical hardware—with an accompanying price tag containing multiple zeros. Don't get me wrong—there are some brilliant hardware products out there from the likes of F5 and Citrix, but I think in a lot of circumstances *HAProxy* is more than capable of doing the job.

The project has a long history, with the first release being made available at the end of 2001. Since then there have been some long release cycles (versions 1.4 and 1.5 took several years each to come to fruition) but the project will still do a reasonable job of back-porting fixes back to these older versions. Version 1.4 still gets critical updates, although I haven't seen anyone running anything older than 1.5.

If you haven't heard of *HAProxy* until now. It's a very capable, very fast and completely free (of course) load balancer and proxy server for TCP and HTTP traffic. The project claims that it's now the defacto open source load balancer and it's difficult to argue otherwise. It's incredibly easy to install by either an OS package of your choice or quite often these days spun up as a container and *HAProxy* helps run some of the largest web properties in the world. I've personally had experience of using *HAProxy* with critical infrastructure where it performed really well (quite often being the most stable part of the platform). I'm a big fan, if that wasn't already obvious.

As for the latest release, the project has claimed it's the cleanest version they have ever produced, with many bug fixes delivered addressing issues with the 1.6 branch as well

HAProxy
The Reliable, High Performance TCP/HTTP Load Balancer

| Quick links | They use it ! |
|---|--|
| Quick News Recent News Description Main features Supported Platforms Performance Reliability Security Download Documentation Live demo Commercial Support Add-on features Other Solutions Contact Us External links Mailing list archives 190kB load-balancing (updated) Contributions Known bugs | <p>Some happy users want to report their experience with HAProxy. Some sites openly confess they use it, think something is inaccurate, or if you'd like to send a few sentences about your experience and have it added to this list.</p> <p>A0 Labs</p> <p>Quoting A0 Labs' CEO Olivier Warin: "A0 Labs is a French hosting company specialized in high availability and performance to their web sites". I would personally like to add that A0 Labs uses HAProxy for all their opensource projects, not just HAProxy !</p> <p>Airbnb</p> <p>Airbnb built SmartStack on top of HAProxy to have exactly the load balancer they needed. The result is a very reliable and performant system.</p> <p>Alibaba / Taobao CDN</p> <p>Taobao's CDN is the world's largest picture CDN, it delivers contents for all online shops including LVS + HAProxy for the load balancing layer, Squid for the cache and their fork of Nginx.</p> <p>Bilibili.com</p> <p>"Bilibili.com uses HAProxy to load balance and provide fault tolerance for 10 million searches per day. Thanks for providing it." - CEO Brendan Sherar</p> <p>BitPusher</p> |

➤ **HAProxy is used by some of the worlds busiest websites, including further down in that list some *ahem* specialist services.**

as changes to improve modularity and tidying up of code. Alas, direct HTTP/2 support was not included in this version (and is now earmarked for 1.8). New features include support for multiple certificates (choosing the best one according to the browser used by the client), OpenSSL 1.1.0 support and improvements around DNS and dynamic updates.

One of the things I like the most about *HAProxy* is the way it isolates itself in a *chroot* jail on startup and drops its privileges so it becomes unable to perform any filesystem access. This is for security purposes and the project claims not to have had any breaches of the software in over thirteen years. It can terminate, initiate and offload SSL/TLS, modify HTTP headers (brilliant for logging requests into an application), provide some protection against DDoS attacks, regulate traffic and perform content based switching.

HAProxy is brilliant at what might be considered its core function, which is to provide load balancing across a set of back-end services. It's easy these days to be blasé about being able to drop servers in and out of a back-end pool, but if you're deploying code to production services and taking a downtime hit (at antisocial hours) simply because you don't have something like *HAProxy* fronting them then I urge you to take a look at it (I'm sure there will be many readers raising eyebrows at such prehistoric sounding setups—but believe me they do still exist).

Of course, the software provides a bunch of load balancing algorithms to suit different scenarios and session stickiness is handled in multiple ways as well. A ton of statistics are provided out of the system as it runs and logging can be modified in all kinds of ways (*HAProxy* provide a handy binary called *halog* which can be used to assist with troubleshooting using the system logs). Finally, it's possible to use Lua as a scripting engine to perform all kinds of complex tasks. Hopefully, I've sold *HAProxy* to you. It really is one of my favourite pieces of open source software. **LXF**

HAProxy Documentation

- Summary
- Available documentation
- Quick introduction to load balancing and load balancers
- Introduction to HAProxy
 - What HAProxy is and is not
 - How HAProxy works
 - Basic features
 - Proxying
 - SSL
 - Monitoring
 - High availability
 - Load balancing
 - Statistics
 - Configuring and managing information
 - Maps
 - ACLs and conditions
 - Content switching
 - Block status
 - Formatted strings
 - HTTP rewriting and redirection
 - Server protection
 - Logging
 - Statistics
 - Advanced features
 - Management
 - System specific capabilities
 - Building
 - String
 - How to get HAProxy
 - Compatible products and alternatives
 - Apache HTTP server
 - NGINX
 - Varnish
 - Alternatives
 - Hardware navigation

➤ **HAProxy has some of the most comprehensive documentation I've seen for an open source project. It's a shame about the logo, though.**



The best new open source software on the planet

LXF Hot Picks



Alexander Tolstoy

squirrels together a big heap of open source applications plucked from the internet's silicon forest to see us through the worst of the cold weather.

Seamonkey » Gnome Recipes » VPaint » Mermaid » Kadu » Inkscape » MPD » Eqonomize » Darktable » 2H4U » Veraball

Internet suite

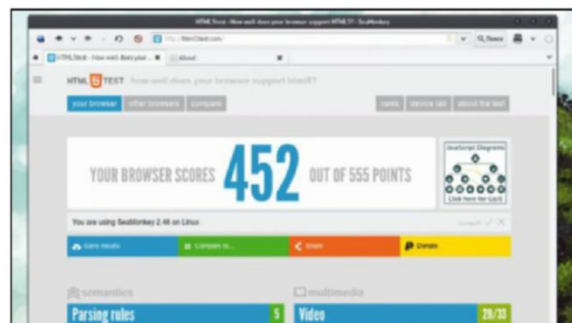
Seamonkey

Version: 2.46 **Web:** www.seamonkey-project.org

The Mozilla project has been the home of the *Firefox* browser since 2004, but it's worth considering that the reason for creating a brand new browser revolved around the bloat of the *Mozilla Suite*, which itself was born from work on the open-sourced *Netscape Communicator*. So, while *Firefox* became the flagship product in Mozilla, the classic *Mozilla Suite* continued on live and was later named *Seamonkey*. Today, *Seamonkey* is one of the most complete, powerful and feature-rich browsers even when put against either *Firefox* or *Chromium*.

Seamonkey Suite consists of a web browser, email client, WYSIWIG web

page composer and an address book manager. Other features can be added as add-ons from the Mozilla store, such as *Chatzilla* (an IRC messenger), *Lightning* (calendar), DOM inspector and many others—many add-ons from the vast Mozilla collection are compatible with both *Firefox* and *Seamonkey* as the browser engine code in *Seamonkey* is synced with the engine from *Firefox*. For instance, the latest



» **Seamonkey's classic and minimal look can be flesh out and made more up to date with some fancy add-ons.**

“One of the most complete, powerful and feature-rich browsers.”

Seamonkey 2.46 has the same Gecko engine as *Firefox 49*, so both browsers render web pages identically, in theory. *Seamonkey* keeps the tradition of the classic web browser, as it was seen back in late the 1990s (e.g. no Australis-like interface). For many users, this design is more appealing than fancy modern browsers, but you can customise *Seamonkey* with a custom set of useful add-ons. This way you can bring back close buttons to tabs, turn the menu bar into a compact button and adjust the application behaviour to your liking. The essential *Firefox* add-ons, including the broad selection of ad blockers and audio/video downloaders are also compatible with *Seamonkey*.

Apart from the browser component, the suite also has a robust email client, which works with all modern mail services, such as Gmail. It's noticeably lighter than *Thunderbird* and doesn't have as many advanced features (e.g. you'll need to manually fill in your incoming and outgoing server details), but it works beautifully. The *Mozilla Composer* is another nice addition for those who want to study HTML basics. Simply design a page in the visual editor and explore the code in the next tab.

Exploring the Seamonkey interface...

Toolbar and menu

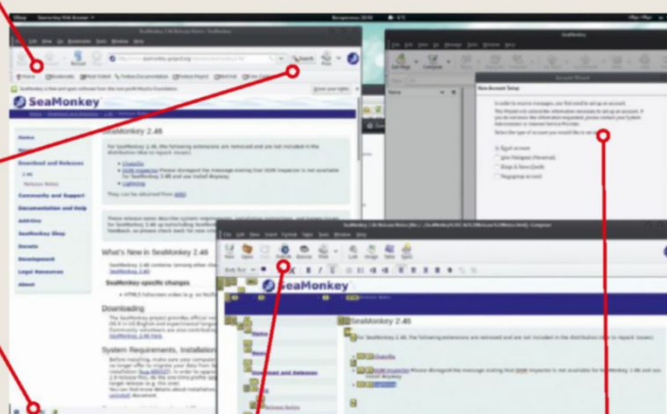
The menu bar is always at your fingertips, although it occupies a bit of extra horizontal space.

The Search button

Type what you want to search for and don't forget to hit that nice 'Search' button.

Switch to another part of the bundle

This tiny button enables you to launch Mozilla composer, email client or address book and each will open in a separate window.



Make your own web page

There are three modes in Composer: WYSIWYG, tag labels and highlighted HTML code.

Sort out your emails

Seamonkey is a lightweight alternative to *Mozilla Thunderbird*, especially for low-end machines.

Cooking application

Gnome Recipes

Version: Git **Web:** <http://bit.ly/GnomeRecipes>

Sometimes we come across productivity applications that are focused on getting you to write down your usual offline household routines, such as shopping lists or personal budget notes, and for many users it probably feels like putting extra effort into something that doesn't need transferring into digital form. Writing down cooking recipes is a similar activity and before we discovered the brand new *Gnome Recipes*, we only knew *Gourmet* and *KRecipes*, which are both essentially note-taking applications with a cooking theme.

Gnome Recipes is a fresh approach to managing recipes, simply because it doesn't just throw a blank editor window at the user, but actually offers real-world recipes right out of the box.

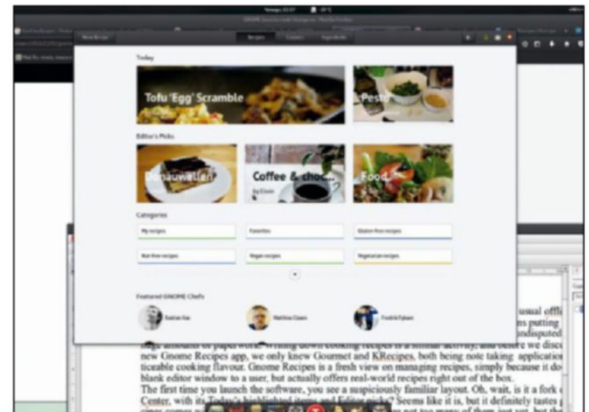
The first time you launch the software, you see a suspiciously familiar layout. It looks like a fork of *Gnome Software Center*, with its Today's

Highlighted Items and Editor picks, but *Gnome Recipes* comes with a set of community made recipes. There aren't too many recipes yet, but the number is constantly growing.

Each recipe has an eye-catching photo of the serving suggestion, the list of required ingredients and cooking directions. You can find out the preparation and cooking times at the upper part of the window and adjust the number of servings and *Gnome Recipes* will automatically update the ingredient quantities in the list below.

The cornerstone of *Gnome Recipes* is the 'New Recipe' button in the first part of the header bar. Developers encourage you to share personal

“Adjust the number of servings and it will automatically update the ingredient quantities.”



► It turns out that some of the core Gnome developers lead a secret second life as chefs.

knowledge for cooking delicious meals, so that people from around the world can populate the application recipes database with tasty dishes from different cultures. User-supplied recipes are processed and moderated via the application's bugzilla. Funnily enough, if someone contributes an inedible entry,

they will soon get a non-resolvable bug in return. So to keep *Gnome Recipe's* open source software free from bugs, you'll need to submit edible recipes!

Drawing and animation app

VPaint

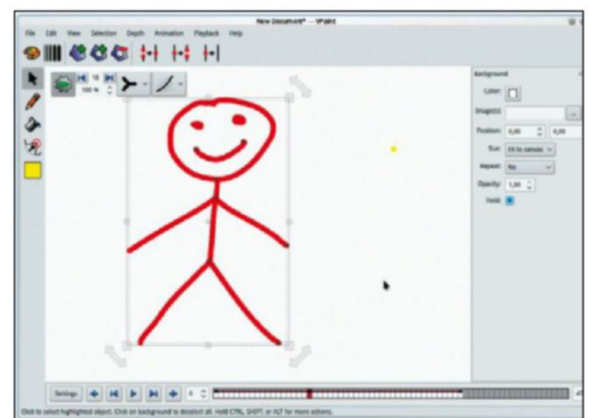
Version: 1.5 beta **Web:** www.vpaint.org

We discovered *VPaint* after futile attempts to find a decent vector graphic editor that would fit in the Plasma5 desktop and offer a bit more than the nearly-abandoned *Karbon* editor from the *Calligra Office* suite. *VPaint* turned out to be an outstanding application in many regards. First, this compact Qt-based program is more than just a vector editor, it's also an animation package, although it shouldn't be compared to such heavyweights as *Inkscape* (see over the page).

VPaint is a proof-of-concept for the academic work of Boris Dalstein and his colleagues from the University of British Columbia. It implements the Vector Animation Complex (VAC), a novel data structure for vector graphics animation, designed to support the modelling of time-continuous topological events. In simple terms, this is a method of

animating hand-made drawings with respect to path connections.

Using the VAC technique you can do advanced things, such as waving wings, rotating figures, heads and even pseudo-3D motion. VAC treats all connected objects as a combination of 'key' elements: vertices, open/closed edges and faces (surfaces). By manipulating these elements you can do simple 3D-modelling via traditional flat vectors and produce many funny things. For instance, *VPaint* enables you to easily draw and animate a dancing man or a flying bird. The bottom bar has frames in its far-right edge where you can adjust and draw an object in



► *VPaint* is perhaps one of the easiest ways to learn animation from scratch.

“VPaint can do some advanced things, such as pseudo-3D motion.”

the first frame. You can then copy and paste it to other frames while changing its shape or position or go with a Motion paste (see the Animation menu item). This way you can make your characters move from a starting point to the destination and *VPaint* will fill intermediate frames automatically. Another point of interest is the View > Advanced (Beta) menu, where you can find some handy but slightly unstable features, such as 3D view, cycle editor and extra options.

Markup language

Mermaid

Version: 6.0.0 **Web:** <http://bit.ly/MermaidMarkup>

Mermaid is an unusual name for Markdown software, which perhaps reflects the way it lures experienced web developers in with its appealing features (alternatively, we can just blame Seth Godin and his much-read *Purple Cow* marketing book). Regardless, we see the average Hotpicks reader as experienced enough to feel bored with the prospect of just another Markdown text editor, so we found something that expands the power of the Markdown language. If you've ever played with Markdown code examples, you'll likely notice that apart from numerous rich text formatting options, it's limited in how it represents schemes, graphics and diagrams. What if we could use something like Microsoft Visio within our Markdown code? Well, it's now possible with *Mermaid*!

Mermaid is written using JavaScript and designed to integrate into web

pages and modern applications, but you can also use *Mermaid* in command-line (CLI) mode. To do so, just install the corresponding module for Node.js:

```
$ sudo npm install -g mermaid
```

You pass the text file to the application and optionally specify extra parameters (`$ mermaid --help` will list all). For example, to generate a PNG with 980 pixels in width you'd use:

```
$ mermaid testGraph.mmd -w 980
```

By default, you'll get a raster PNG image as the output, but you're free to export your code to SVG and optionally use custom CSS files to control the look and feel of your graphs.

The .mmd file should contain a proper Markdown-like syntax that



➤ **Mermaid enables you to add some code for a Gantt diagram to your Markdown document?**

matches specific Mermaid rules. You can draw simple graphs, subgraphs, flowcharts, sequence diagrams and any combinations of all the above and get nicely rendered *Visio*-like graphics immediately in your web browser.

Mermaid is also supported as an extension in CuteMarkEd (<https://cliclose.github.io/CuteMarkEd>), a handy visual Markdown editor that renders your code with zero delay. To

use this you need to wrap your *Mermaid* code in `~~~~mermaid and ~~~~` and enable the feature in the Extra > Diagrams support menu.

“What if we could use something like Microsoft Visio within our Markdown code?”

Instant messenger

Kadu

Version: 4.2 **Web:** www.kadu.im

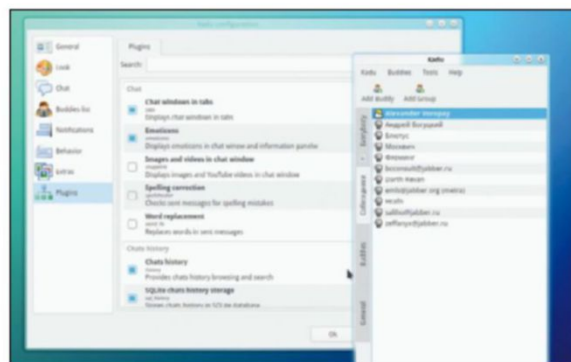
Updated *Kadu* versions have appeared several times lately on linux-apps.com, so it was easy to pick it up for HotPicks. This instant messenger (IM) is outstanding in many ways, and there's a good chance it's one of those overlooked applications you probably need. Not only because its rivals are weak: KDE *Telepathy* (KTP) suffers from many bugs, such as a broken OAUTH2 plugin that makes connecting to many Jabber services impossible and the likes of *Kopete* haven't received any noticeable updates for a long time.

Kadu is a KDE-centric client for Gadu-Gadu, a popular Polish IM protocol, but you don't need to be a member of Gadu-Gadu network as *Kadu* also supports Jabber/XMPP.

At the first look, *Kadu* is a typical IM client and while it's very robust, it's also remarkably comprehensive and

feature-rich. However, there are very few binary packages for *Kadu*, so you'll probably need to build it from source and that's the point where you'll see how large its scope is. *Kadu* has numerous plugins and extras from MPD daemon integration (which lets your buddies know what you're listening to right now) to an optional Ubuntu Unity indicator, word replacement module, images and videos integration for the chat window.

Kadu has a lot of dependencies and it takes some time to install them all before the usual `$ cmake ..` command finishes without errors. It takes 10-20 minutes more to compile the



➤ **There are many ways to customise your Kadu installation and make it very comfortable to use.**

application itself (`$ make`) and extra effort to install it to a system-wide prefix (such as `/usr`). After you take a breather, you'll also need to rebuild your distro's `libxmp` package in order to prevent *Kadu* from crashing when you add an XMPP account.

However, *Kadu* is stable and well-tested in other regards and in return for your efforts you get a perfectly versatile IM client with lots of bells, whistles and extras that cater for many tastes. *Kadu*'s interface is also very clean and polished and sports sensible placement of buttons, drop-down lists and categories.

“In return for your efforts you get a perfectly versatile IM.”

Vector editor

Inkscape

Version: 0.92 **Web:** <https://inkscape.org>

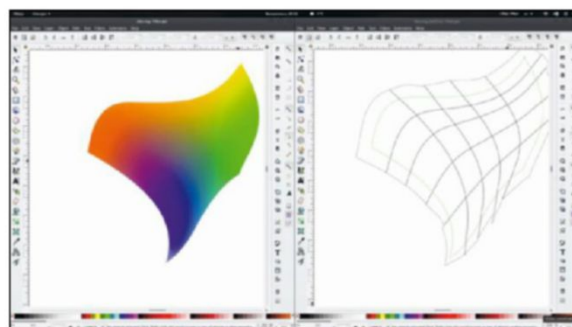
The last time we reviewed *Inkscape* was back in 2015 for version 0.91 [See HotPicks, p60, LXF197]. *Inkscape* is a state-of-the-art vector graphics editor for Linux that easily outperforms its many rivals. Its releases are not frequent and the project is very important for anyone who likes to keep a close watch on open source software focussed on creative work. *Inkscape* is widely used in publishing for producing high-quality layouts and graphics for print and the web, in a similar way to *Adobe Illustrator* and *Corel Draw*. Many Linux-related artwork is also made with *Inkscape*, including multitudinous icon sets and even window decorations (e.g. the Aurora theme engine).

The new *Inkscape 0.92* was released after almost two years of development and there are some bold changes worth noting. The biggest innovation is mesh gradients—the new tool that enables

you to build a mesh that follows the shape of an object and defines colours for every node of that mesh, which means that you can now draw photo-realistic vector illustrations with smooth colour transitions. *Inkscape's* developers have also made the application more suitable for making graphics for the web: the new version 0.92 adds support for over a dozen SVG and CSS3 properties and complies with industry standards defined by the official SVG standards committee.

There are also new tools and modes for drawing, such as Spiro Live, BSpline, and Roughen for path effects; a new tool that non-destructively cleans objects from unnecessary nodes; the

“You can now build a mesh that follows the shape of an object.”



► A quick illustration of how to achieve smooth colour transitions within a vector drawing in *Inkscape*.

new interactive Perspective/Envelope and Lattice Deformation 2 tools and an endless number of small changes. To maintain consistency with the CSS standard, *Inkscape's* default resolution has changed from 90dpi to 96dpi. If you have existing graphics made with a previous *Inkscape* version, the new *Inkscape 0.92* will prompt you to convert resolution to the new default.

Starting from this version, *Inkscape* now uses the *CMake* build system that accurately checks for build dependencies and is a bit faster when it comes to compiling the code with `make`. Hopefully, by the time you read this, your distro should have the new version of *Inkscape* in its repos.

Music daemon

MPD

Version: 0.20 **Web:** www.musicpd.org

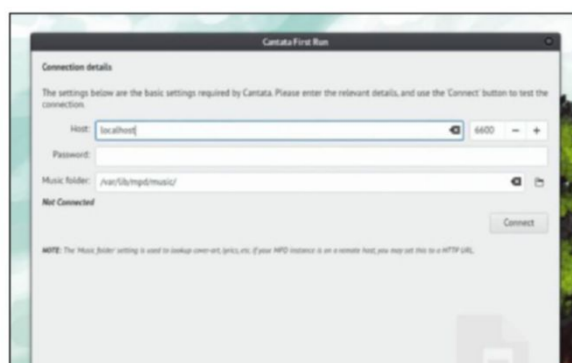
There are different approaches to listening to digital music. Some people trust online streaming services and pay for subscriptions, others prefer the tried-and-tested method of keeping their local music library. The latter still has some issues, such as an inability to access your music collection from other devices. But there's a work around for the curious-minded: you can store your music on the local network drive (NAS) and access it remotely.

MPD is a very useful tool for everyone who has a home LAN with at least two devices. MPD, a music player daemon, isn't a player but a core part of the client-server architecture for accessing music, it can play your files, but you should have a client application to access the daemon. There are many MPD clients, including ones for the command-line (e.g. *mpc/bmpc*),

pseudo-graphical text-based ones (e.g. the ncurses-based *ncmpc*) as well as graphical clients (e.g. *Ario* and *GMPC* etc). Even if you're not up to splitting your media infrastructure between separate devices, you can still happily use MPD and any of its clients on your localhost machine.

The new MPD 0.20 can now read ID3 and APE tags from remote files (HTTP/NFS/SMB protocols are supported) and features much improved sound quality for streaming high-definition audio. Specifically, MPD now supports playing and streaming uncompressed 16-bit Linear PCM sound (aka L16), as well as DSD_U32

“This update makes MPD a feasible solution for audiophiles.”



► MPD enables you to access your music collection from other devices—in this case a web server.

(Direct Stream Digital) and Wavpack-DSD. This update makes MPD a feasible solution for audiophiles who demand the highest quality of uncompressed sound. For instance, the Wavpack format is often considered as the most balanced one: it delivers the original lossless quality together with decent file size optimisations.

If you're fond of listening to music and exploring various players for Linux, updating to MPD is recommended. However, keep in mind the use cases where MPD plays at full power, such as streaming music over a network, especially in a multi-channel mode.

HotGames Entertainment apps

Arcade game

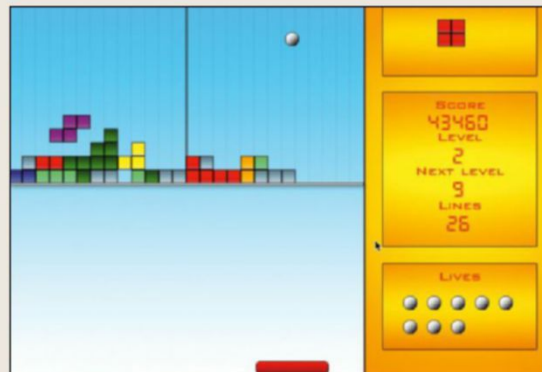
2H4U

Version: 1.3 **Web:** <http://bit.ly/2H4Ugame>

Being able to use both hands with (almost) equal ease can really pay off, especially in sports, arts and music. But we've found a game that uses that ability, even though it could be too hard for you (which explains the name!). Meet *2H4U*, a combination of *Tetris* with a wall-breaker! This is a fun, small game that wants you to build a brick tower from familiar blocks of different shapes, but with a moving platform at the bottom of the screen. The *Tetris* part of the game is driven by the WASD keys, while the platform for breakout can be moved with arrow keys. When the game starts you have limited leeway for about a minute. When you build the first *Tetris* lines, the breakout section is inactive, but later on the ball starts

flying and you need good coordination to deal with both parts of *2H4U*. If you are like 90% people that are right-handed, you'll need to develop the skill of playing with both hands.

The goal of the game is to score more points, but the more you play, the more strategy becomes important. In one respect, you want your ball to penetrate through the line of bricks and stay there for a long period of time in order to leave a breakout platform unattended and give a little rest to one of your hands. But that means that you need to construct a proper block



» **Save your time: play two games at once!**

configuration, which isn't too small to give a ball enough obstacles to jump off inside the tower, but at the same time not too cluttered for accidentally losing the game when a new brick has no space to land.

We found *2H4U* fun to play as it melds the benefits of both game elements together which results in a perfect brain workout. Frankly, the name of the game seems a little overblown: but you can always make it harder by swapping hands.

“The more you play, the more strategy becomes important.”

Ball arcade

Veraball

Version: Git **Web:** <http://bit.ly/Veraball>

Last time, we discovered the fabulous *Mousepond* and this time we're playing another Godot-powered game called *Veraball*. The first time we saw it we were reminded of *Neverball* [see Hotpicks, p64, LXF193], another 3D game where you control the ball inside a futuristic maze.

In fact, *Veraball* is very similar to *Neverball*, so it is worth comparing the two. The main difference between the two is the way of controlling the ball. In *Veraball*, you can change the point of view with your mouse and just move the view around to look at the world from a different angle. When you move your mouse, you just control the camera not the ball. To roll the ball you need to use the WASD or arrow keys. The ball's behaviour and momentum

reflects the ball's heaviness and it takes time to accelerate or slow it down than in *Neverball*. As with *2H4U* (above), you need to develop good coordination in order to control the ball effectively.

The current camera view is the reference position for keys for controlling the ball, which means that you have two ways to control the moving direction: you can either use only the keys and change camera angle only when necessary (e.g. when a part of the maze obscures the view) or use only forward and back keys and adjust direction with your mouse. The difficulty is not to muddle your hands.



» **Despite the stone-grey colours, the graphics detail is high and the physics are stunning.**

It takes time to play *Veraball* without losing the game every 30 seconds, because the first level is quite challenging. You need to go through narrow paths, climb chutes and pass through fragmented bridges—all within 60 seconds. Despite the difficulty, the process is very fun and entertaining.

The game source code doesn't need any compilation. You have to get the *Godot Project Manager* (<https://godotengine.org>), run it and import the downloaded game. Once you select the game's engine.cfg file, *Veraball* will appear in the project list.

“Go through narrow paths and climb chutes—all within 60 seconds.”

Financial application

Eqonomize

Version: 1.0 beta 1 **Web:** <http://bit.ly/Eqonomize>

Linux Format's obsession with personal finance management continues. Meet *Eqonomize*, a compact Qt-based tool for maintaining your home budget. Technically, it does nearly the same job as *Skrooge* or *KMyMoney* [See Roundup p24 LXF215], but it does so by approaching budgeting with a different approach.

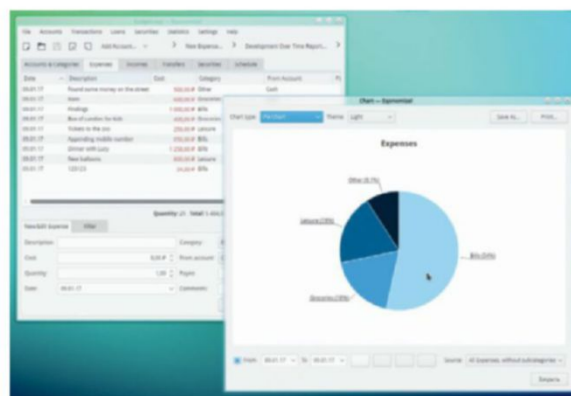
The first thing you may notice after launching *Eqonomize* is that it uses your screen space efficiently. There's no need to keep the window maximised as everything fits into a very compact window just fine. Another appealing difference is that *Eqonomize* is ready to record your transactions right away.

There are no wizards and account settings at the start-up—by default you get a ready-to-use blank account set up to use the currency of your country (based on your locale). The interface of *Eqonomize* has six tabs, of which

Accounts & Categories always loads first by default and provides the overview of all your finances. Next, there are lists of expenses and incomes, transfers, securities and a schedule.

Eqonomize supports double-entry bookkeeping and enables you to evaluate your ledger at any time (Accounts > Ledger). The application also offers easy-to-understand dialogs for conducting split transactions (several transactions as one record), managing loans, debts and securities and scheduling recurring transactions etc. After a while, when you collect some money flow statistics, you can view reports and charts.

“Eqonomize is ready to record your transactions right away.”



► When using *Eqonomize*, a colourful pie-chart with all your expenses is just a few clicks away!

Eqonomize has just two types of statistical reports: development over time and categories comparison, but these are perfectly good enough to get a picture of what is going on. There are more advanced features in *Eqonomize* that make it more like a professional tool (managing stocks, bonds and shares), so this application can be used beyond small households as well. All the information is stored in human-readable XML format and can be optionally exported to the industry-standard QIF format.

Photography tool

Darktable

Version: 2.2.0 **Web:** www.darktable.org

Generally speaking, if you want to get the best results out of a digital SLR camera you'll want software to process the files. While there's very little sense in enhancing images taken on mobile devices (usually the default JPEGs are the best that your smartphone camera can produce), digital SLR cameras can record all pixels directly from their sensors to the RAW file format.

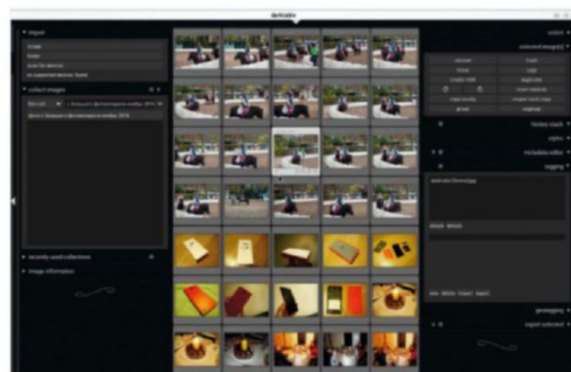
There are many applications in Linux that can process RAW files and *Darktable* was one of the first open source RAW tools that appeared for Linux, and, while longevity doesn't always imply perfection, we believe that every Linux photo enthusiast should be aware of some of *Darktable*'s very useful features.

Darktable works in two modes: lighttable and darkroom. The first mode resembles an image viewer with thumbnail support—you select a

directory with a corresponding tool on the left edge and wade through dozens of photos in order to sort out the best shots. Once you double-click an image, you're taken to darkroom mode where the magic begins.

A set of colour correction tools on the right edge of the window enable you to fix colours and produce better quality images than the default camera optimiser. You can tune shadows and highlights, fix white balance, sharpen or apply a lens mask to fix distortion and as long as RAW stores more data and preserves both details and colour dynamic range, most of your changes will be non-destructive. Based on our tests, *Darktable* is very robust in

“Every Linux photo enthusiast should be aware of some of Darktable's useful features.”



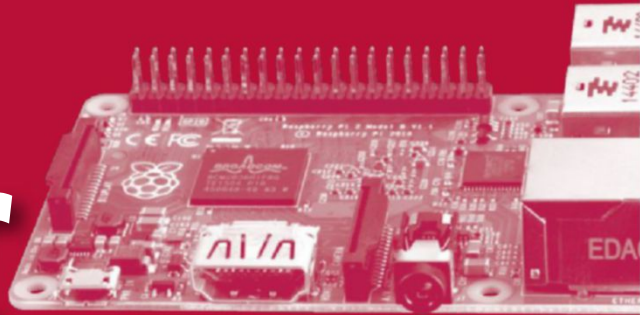
► *Darktable* can work with all image formats, though the best results can be achieved with uncompressed images.

handling heavyweight RAW files, it easily applies all your changes with little or no delay.

The only thing you can't do inside a darkroom is exporting. In order to save or export your work you need to get back to the lighttable and use the menu again on the right edge. This time there will be tools for selecting images, editing styles metadata tags and exporting.

Building *Darktable* from source probably isn't necessary—it's been picked up by most distros and the packages are often kept up to date.

LINUX FORMAT Pi user



Giving you your fill of delicious Raspberry Pi news, reviews and tutorials

RACHEL RAYNS
Maker and artist
working for The
Pi Hut.



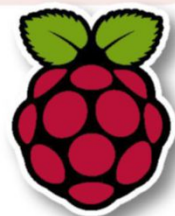
Welcome...

Hello, I'm Rachel Rayns. I'm a maker and artist based in the east of England. I previously worked at the Raspberry Pi Foundation and now I'm making products with The Pi Hut.

My favourite thing about the Raspberry Pi computer and the maker movement in general, is how it can foster personal empowerment in people who don't know they're makers—yet! Whenever I tell new people about the Pi, I can't help but feel like I'm sharing the secret of some superpower. Things don't have to act the way large electronics companies have decided—you can change things! You can choose! Want your hallway light to turn on when you pull up in your driveway? Make it! Grandma always falls asleep with the TV on? Make something to solve it—and make it with Grandma. Your tech can make someone's life more efficient or just make them smile.

Developing making, hacking and coding skills allow people to change things in their environment to work best for them. Rapid prototyping, small-scale manufacturing and close relationships between recipient/designer/manufacture (sometimes all three are you!) means bespoke tools can be created when traditional manufacturing would never allow them to. Most importantly, these skills enable people from varied backgrounds to get involved in production. The Pi computer is so low cost and has mountains of good, free documentation, it makes it a really accessible tool for everyone, not just those with a computer science or engineering background.

The Pi's Digital Making Curriculum



Providing a structure learning path for everyone and anyone looking to improve their skills.

The Raspberry Pi Foundation has done a tremendous amount for education even in the short time it's been around. To help students, teachers and everyone take the right steps to enhance their skill and learn new abilities the Pi Foundation has created its own Digital Making Curriculum.

The aim of this in its own words is that "you start with certain skills and knowledge and then, with guidance, practice, and understanding, you gradually progress towards broader and deeper knowledge and competence." The curriculum breaks down the process of becoming a full 'Maker' into digestible stages suggesting projects and skills

that people should consider attempting to help them progress.

The Pi Foundation goes to great pains to emphasise this doesn't replace any formal curricula but is there to help and support such things; including those people running informal maker spaces and JAMs. Discover more at: www.raspberrypi.org/curriculum.

» We were always told education would 'arm' us well.



Compute 3 Blobless Pi

Mini Pi 3 is here.

The original Compute Module was launched back in April 2014 now an updated Compute 3 brings the power of the quad-core Raspberry Pi 3 processor to the largely industrial installations it's aimed at, such as the NEC TVs. Running at 1.2GHz, 1GB of memory and an optional 4GB of onboard eMMC flash. www.raspberrypi.org/blog/compute-module-3-launch



» So small and soooo cute!

Open firmware moves.

The dream is for a full open ecosystem: from the boot loader to the kernel and desktop. Some might say Raspberry Pi's failing is the various binary blobs required to boot it. So the rpi-open-firmware (or Blobless Linux) project is one to keep your eye on. <http://crma.cc>



» This is what freedom looks like.



Rainbow HAT

Les Pounder jumps on the Internet of Things wagon and learns how to build his own device.

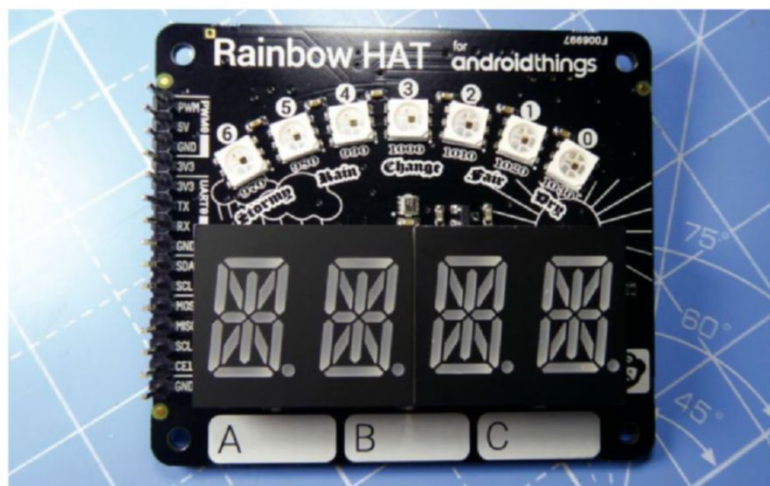
In brief...

» An Internet of Things board designed for use with the new Android Things platform. But this board is equally at home being used in a simple Python-powered project, because it comes with an extensive array of inputs and outputs. The Rainbow HAT can be bought on its own or as a full kit, which includes a Raspberry Pi 3.

The Internet of Things (IoT) is all around us. Baby monitors, refrigerators, smart ovens and central heating systems are now part of the internet and, when they are not part of a botnet, they enable us to control all these devices remotely, ensuring our rooms, or our food, are all warm when we get home. But how can you take your first steps with IoT? The Rainbow HAT board for the Raspberry Pi 3 is one contender that offers a hardware platform that's compatible with the new Android Things software, which uses Android and compatible devices to create IoT devices.

The board comes with a plethora of components, including capacitive touch inputs, APA102 multicolour LEDs, a piezo buzzer, four 14-segment alphanumeric LED displays, and a BMP280 temperature and pressure sensor. It can be further extended and used with external devices thanks to breakout pins for PWM (for servos), UART, I2C and SPI. Using the Rainbow HAT, it's possible to bridge the world of software and hardware, creating our own IoT devices – but how easy is it? Well, the tagline for this project is 'If you can build an app, you can build a device,' and although this is generally true, it is a little more involved.

To use your Raspberry Pi 3 with Android Things, you need to download a special operating system, which effectively turns your Pi into an appliance; a device to perform one task rather than a full-blown computer. In



» Fitting on top of the Raspberry Pi 3, the Rainbow HAT is an unassuming board that offers multiple methods of input and output.

order to program the Pi, we require the use of the *Android Studio IDE*, installed on another computer. We then connect remotely to the IP address of the Pi using the *Android Debugging Bridge (adb)*, and push the application to the Pi, where it will run after a few minutes. Android applications and, in this case, devices, are written using a Java API framework. There are plenty of example projects to illustrate how the Android Things platform and Rainbow HAT can be used together. We tested the weather station device project, and within 15 minutes we had a working weather station, outputting data collected from the BMP280 sensor to the board's LEDs and to our screen. As well as using common Google services, the Android Things platform supports working with the GPIO of the Pi 3 and other devices, namely the Intel Edison and NXP Pico development boards. However, the Rainbow HAT is only compatible with the Raspberry Pi 3.

But what if using the *Android Studio IDE* is too much for you? Can you still get value for money from this board? Yes. The Rainbow HAT can also be controlled using Python, and thanks to a relatively painless install and comprehensive Python library, we can control all of the components present on the Rainbow HAT. So this board can be used as your own Python-powered Pi 3 IoT device – all you need to do is

incorporate some extra libraries, such as Weather API, Twitter API and so on, and your project can display data on the 14-segment LED displays, monitor temperature and pressure using weather sensors, and compare them to Weather API data for your location.

The Rainbow HAT board is versatile, and in the experienced hands of an Android developer, is a powerful platform when combined with Android Things. But it's not for everyone. If you want LEDs and temperature sensors, then the Sense HAT is an informed purchase. But if you want to create IoT devices using sensors, inputs and outputs in one handy package, the Rainbow HAT is an ideal product to drop into your next project. **LXF**

Features at a glance



Expansion

With access to protocols such as PWM, UART, I2C and SPI, users can add extra sensors and outputs.



Buzzer

The piezo buzzer can be used to create music, and the Python library can be used with MIDI notes.

LINUX Verdict

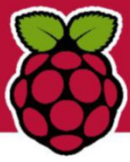
Rainbow HAT

Developer: Pimoroni
Web: <https://shop.pimoroni.com>
Price: From £24 to £80

| | |
|-------------|------|
| Features | 9/10 |
| Performance | 9/10 |
| Ease of use | 8/10 |
| Value | 9/10 |

» An impressive package for both the beginner and the expert, thanks to Python and Android Things interfaces.

Rating 9/10



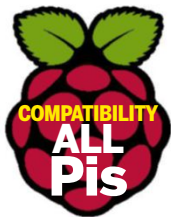
GPIO Zero: Memory game

Les Pounder shows us how to use GPIO Zero, buttons and LEDs to create your own simple 1980s inspired game.



Our expert

Les Pounder works with the Raspberry Pi Foundation delivering its Picademy training. He also likes to tinker with electronics and toys and blogs at <http://bigl.es>



Quick tip

The GPIO Zero resources are constantly receiving updates thanks to a lively development team. In recent releases we've seen servo support and ultrasonic sensors, pulse width modulation (PWM).

The *Simon* memory game was particularly popular when we were growing up. This was an electronic game that used LEDs and big buttons to challenge players to remember and repeat a complex sequence. We're going to make our own version using a Raspberry Pi.

In this tutorial, we'll use coloured LEDs (one blue, green, red and yellow) and four momentary buttons/switches which we'll assign to each LED and we'll be using Python and GPIO Zero to code the project. You can use any model of Pi, but you'll need the latest Raspbian Pixel Release. You'll also need four 220 Ohm resistors (orange-orange-brown-gold), a half-size breadboard, 10 male-to-female jumper wires and four male-to-male jumper wires. For instructions on how to connect your circuit refer to the diagram. (All of the code for this project and the diagram can be found at <https://github.com/lesp/LXFMemoryGame/archive/master.zip>.)

All of the software necessary for this project is pre-installed on Raspbian/Pixel. So to start coding we need to click on the main menu and go to Programming > Python 3 Editor. Once the editor opens we need to click on File > New to open a new blank document and click on File > Save in the blank document and call your file **memorygame.py**.

Our first block of code involves importing modules that are used to provide extra functions for our project.

```
import time
import random
from gpiozero import LED, Button
```

Here we import the **time** library so that we can control the pace of our project. We import **random** to introduce random numbers into our project. Last, we import two sections of the GPIO Zero library, which deal with LEDs and buttons. GPIO Zero has a class called **LED** which is used to control the GPIO pins as outputs. We tell the class which pin is connected to the long leg of each LED. Here we create four objects that store the location of each LED for each colour:

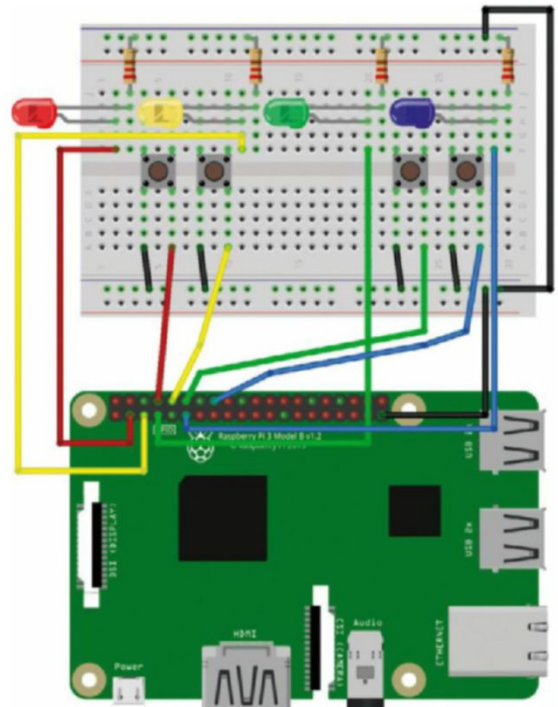
```
red_pin = LED(2)
yellow_pin = LED(3)
green_pin = LED(4)
blue_pin = LED(17)
```

There is also a class to work with GPIO pins as inputs and it is called **Button**. Here we use it to create objects for inputs.

```
red_button = Button(14)
yellow_button = Button(15)
green_button = Button(18)
blue_button = Button(23)
```

To control our LEDs we are going to create a function, called **lights** that has two additional arguments:

```
def lights(colour,duration):
```



► The project's circuit has many components but it's easy, just trace each line from the Pi to the component.

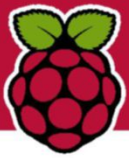
A function is a block of code that can be executed by calling its name. The arguments for the function are for the LED colour and how long it should be displayed:

Inside of the function, all lines of code are automatically indented four spaces. We create a conditional test that will use the colour argument to turn on the correct coloured LED.

So if we used **"red"** then the red LED will turn on.

```
if colour == "red":
    red_pin.on()
    time.sleep(duration)
    red_pin.off()
    time.sleep(duration)
elif colour == "yellow":
    yellow_pin.on()
    time.sleep(duration)
    yellow_pin.off()
    time.sleep(duration)
```

Afterwards, we use the **sleep** function with the duration to control how long the LED is on for. We then turn the LED off and again pause for the duration. We do this for each LED, here (above) we see the first two blocks of code that control



Components explained

All electronics devices are built using components. These come in many forms, some small devices require surface mount components which are soldered on top of a board. Some larger devices use through-hole one, which are typically bought from hobbyist stores. These are inserted into holes on a board and soldered to pads on the opposite side. No matter the component, it's built into a circuit.

Building the circuit for this project requires quite a few through-hole components. First, we use LEDs (Light Emitting Diodes) that illuminate when current is passed through them. We use LEDs as a method of output, the light sequence that a player must remember in our game. To limit the current that each LED draws we use a resistor, if we didn't do this then the LEDs would quickly, if not instantly burn out. The last

component we use are four momentary switches/push buttons, which when pressed will complete a connection on our breadboard. We use them to repeat the LED sequence into our code. With each button relating to an LED colour. All of these components are connected using a breadboard, a plastic board that has metal connectors that run inside and enable components to connect in a temporary circuit.

"red" and "yellow", the code for "green" and "blue" is the same as "yellow". We've truncated the code.

Now out of the function we create a list; a data structure that's used to store multiple pieces of data, each with their own place, called an 'index' in the list. Here we call the list `leds` and store the colours of each LED.

```
leds = ["red", "yellow", "green", "blue"]
```

To check that our LEDs are working and to signify the game is ready we use a for loop that will iterate over each colour in our `leds` list and call our `lights` function.

```
for led in leds:
```

```
lights(led,0.3)
```

We now enter into an infinite loop that will run our game, all of the code inside of the loop is indented four spaces.

```
while True:
```

Inside the loop we create a trigger to start the game. This trigger is pressing the button assigned to match our "red" LED. When pressed, it starts a countdown using `print`:

```
red_button.wait_for_press()
```

```
print("Standby for light sequence!!!")
```

```
time.sleep(1)
```

```
print("3")
```

```
time.sleep(1)
```

```
print("2")
```

```
time.sleep(1)
```

```
print("1")
```

```
time.sleep(1)
```

```
print("GO!!!")
```

As soon as the countdown is over we immediately shuffle the contents of the `leds` list we created earlier. This gives us a random list of colours for our game.

```
random.shuffle(leds)
```

We then play the lights sequence to the player using a for loop that will iterate over each entry in our `leds` list and will use our `lights` function to turn on each LED.

```
for led in leds:
```

```
lights(led,0.2)
```

We then create another list, called `answer` but this time we leave it empty, we shall fill it later.

```
answer = []
```

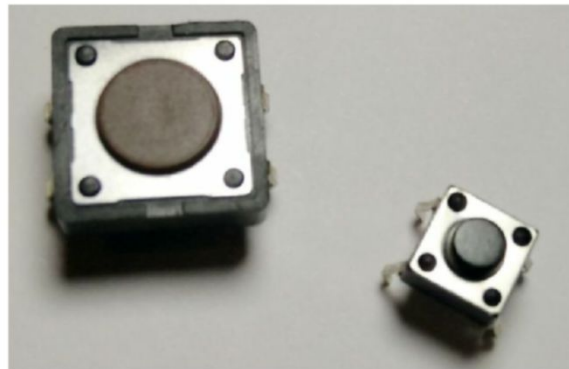
We now enter another while loop, inside the original while True loop. This new while loop will run for as long as the length of our answer list is less than four items long.

```
while len(answer) < 4:
```

The goal of this loop is to capture button presses corresponding to the colours of the LED sequence:

```
time.sleep(0.3)
```

```
if red_button.is_pressed:
```



› Momentary switches/push buttons come in various sizes, but generally if they look like these then they will fit into your breadboard.

```
answer.append("red")
```

```
lights("red",0.1)
```

```
print(answer)
```

```
elif yellow_button.is_pressed:
```

```
answer.append("yellow")
```

```
lights("yellow",0.1)
```

```
print(answer)
```

To capture the sequence, we use a delay of 0.3 seconds to ensure the LEDs have stopped flashing. Next, we use another conditional statement test that checks to see the status of our buttons. Once a button is pressed the corresponding colour is appended to our answer list and the correct colour flashes to indicate our selection. We also print the answer to the shell. We now come out of the if conditional test and while len loop, but remain inside the main while True loop.

Our final section of code is another if conditional test, but it compares the randomly chosen sequence of LEDs with the player's answer. If correct, the player wins and we use a for loop to blink all of the LEDs.

```
if leds == answer:
```

```
print("YOU WIN!!!")
```

```
for blink in range(10):
```

```
lights("red",0.1)
```

```
lights("yellow",0.1)
```

```
lights("green",0.1)
```

```
lights("blue",0.1)
```

But if the player gives an incorrect answer they lose the game and only see a blinking red LED.

```
else:
```

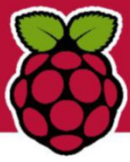
```
print("YOU LOSE!!!")
```

```
for blink in range(10):
```

```
lights("red",0.1)
```

With that our project is complete. Save your work and click on Run > Run Module to launch the game. To make the game harder, shorten the duration in the lights function. **LXF**

» Love your Pi more: Subscribe and save at <http://bit.ly/LinuxFormat>



AUI Suite: Voice control your Pi

Bend your Raspberry Pi to your will by issuing vocal commands.



Our expert

Nate Drake is a freelance technology journalist that specialises in cybersecurity and retro tech.

There's a scene in *Star Trek IV: The Voyage Home*, where Scotty and Dr McCoy are offered the opportunity to use a computer and are somewhat bemused when it fails to respond to voice commands. The film was made over 30 years ago and voice recognition has taken huge leaps forward since then. The Raspberry Pi has not been left out, and with the help of an inexpensive USB microphone and some suitable software, you too can have your Pi search YouTube, open web pages, launch applications and respond to questions, simply by speaking.

The Raspberry Pi has no built-in soundcard or audio jack, so you need a USB microphone or a webcam with built-in microphone for this project. We tested the software using a Microsoft HD-3000 webcam, but any compatible device will do. Visit http://elinux.org/RPi_USB_Webcams for a full list of Pi-compatible webcams if you don't already have one, but be sure that whatever device you choose has an integrated microphone.

If you only have a microphone with an audio jack, try searching Amazon or eBay for an inexpensive USB sound card, which plugs into the USB port at one end and has an output for earphones and a microphone at the other.

There's a number of speech recognition programs for the Pi. For this project, we're using Steven Hickson's *Pi AUI Suite*, because we've found it to be powerful as well as extremely easy to set up and configure.

Once you follow the steps in the tutorial, you will be able to start the installer. The *Pi AUI Suite* gives you a choice of a number of programs to install. The first question you are



► If you have a mic with an audio jack, you may be able to use a small USB soundcard to make it work with the Pi.

asked is whether it should install the dependencies. These, quite simply, are the files the Raspberry Pi needs to download for voice commands to work, so select y and press Return to agree to this.

Getting started

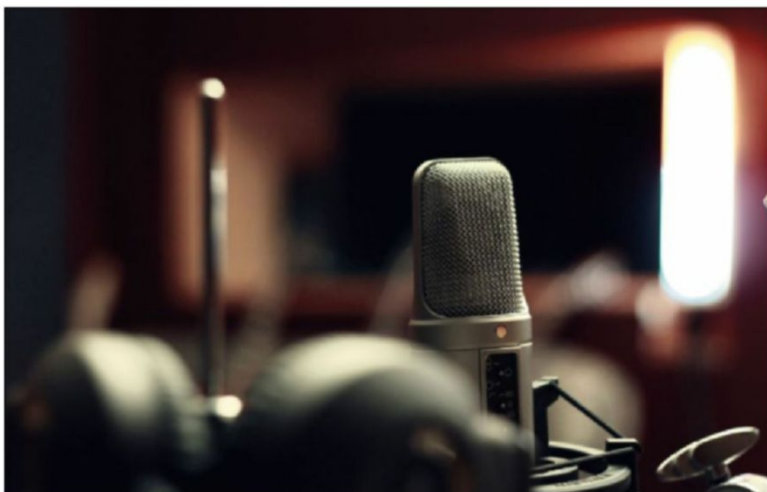
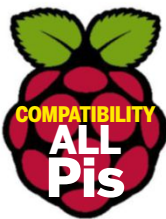
Next, you are asked if you want to install the *PlayVideo* program, which enables you to use voice commands to launch and play video files. If you choose y, you're asked to specify the path to your media files, for example, `/home/pi/Videos`. Note that upper-case letters are important here. If the path is invalid, the program warns you.

You're then asked if you want to install the *Downloader* program, which searches for and automatically downloads files from the internet for you. If you choose y here, you're asked to provide settings for host, port, username and password. If you aren't sure of these, press Return for now to choose the default options in each case.

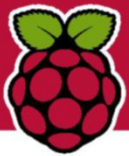
The next program is *Google Text to Speech Service*, which you may wish to install if you want the Pi to read out the contents of text files. In order to use this service, the Pi needs to be connected to the internet, because it connects to Google's servers to 'translate' the text into speech and then plays an audio file with the Pi's media player.

If you decide to install this, you will need a Google account. The installer asks you to enter your username. Do so, then press Return. You're then prompted for your Google password. Enter this and press Return again.

The installer also offers you the chance to install *Google Voice Commands*. This uses Google's own speech recognition service. Again, you're asked to provide your



► See whether you can find a USB microphone or webcam.



Basic voice commands

Once installation of *Pi AUI Suite* is complete and you have run `sudo voicecommand -c` to set it to listen, you need to prime it with a keyword. By default, this is 'Pi', but feel free to alter this to something easier, such as the word 'Jessie'.

Next, try out a few of the built-in voice commands:

» **YouTube** Saying 'YouTube' and a video title automatically loads a

full-screen video of the first relevant YouTube clip. This is similar to Google's "I'm feeling lucky". Say 'YouTube' and the name of the video in which you're interested, e.g. 'YouTube fluffy kittens'.

» **Internet** Saying the word 'internet' launches your web browser. By default, this is the Pi's built-in browser *Midori*, although you can change this.

» **Download** Saying the word 'download' plus a search term automatically searches the Pirate Bay [careful now—Ed] website for the file in question, for example, you could say 'Download Ubuntu Yakkety Yak' to get the latest version of the Ubuntu 16.10 distro.

» **Play** This command uses the built-in media player to play a music or video file, e.g. 'Play

mozartconcert.mp4' would play that particular file located in the media folder you specified in setup, such as `/home/pi/Videos`.

» **Show me** Saying 'show me' opens up a folder of your choice. By default, the command doesn't go to a valid folder, so you need to edit your configuration file to a valid location, for example `show me=/home/pi/Documents`.

Google username and password to continue. Whether or not you choose the Google-specific software, the program also asks you whether you want to install the YouTube scripts. These tools enable you to speak a phrase such as 'YouTube fluffy kittens', which will trigger a search and see YouTube play a relevant video from the results.

Finally, the program gives you the option to install *Voicecommand*, which contains some of the more useful scripts, such as being able to launch your web browser by saying the word 'internet'.

The program asks you if you want to let *Voicecommand* set itself up automatically. If you experience an error at this stage, follow Step 3 of the walkthrough (see the last page).

Pi's master's voice

Once the *Voicecommand* program is installed, you may wish to make a few basic changes to the setup before fine-tuning your configuration. To do this, open Terminal on your Pi or connect via SSH and run the command

```
sudo voicecommand -s
```

You are asked a series of yes/no questions next. The first question asks whether you want to permanently set the continuous flag. In plain English, the *Voicecommand* program is asking whether, each time you run it, you want it to continuously listen for your voice commands. Select y for now.

Next, you are asked if you want the *Voicecommand* program to permanently set the verify flag. Selecting y here means the program expects you to say your keyword (by default, the word 'Pi') before responding to commands. This can be useful if you want to set the Pi to listen continuously and don't want it to act on everything you say.

The following prompt asks if you want to permanently set the ignore flag. This means that if *Voicecommand* hears a command that's not specifically listed in your configuration file, it tries to look for a program in your installed applications and run it. For instance, if you say the word 'leafpad', which is a notepad application, *Voicecommand* searches for and runs this even if not specifically told to.

We don't recommend you enable this feature. Because you're running *Voicecommand* as a SuperUser, there's too much risk that you could inadvertently tell the Pi a command that could harm your system. If you want to set up extra applications to work with *Voicecommand*, you can edit the configuration file in each specific case. (See *Voice Command Tweaks*, over the page.) *Voicecommand* then asks you whether you wish to permanently set the quiet flag, so it

doesn't give a verbal response when you speak. Choose y or n as you see fit.

Next, you're asked if you want to change the default duration for speech recognition. You should only change this if you're finding the Pi is having trouble hearing your commands. If you choose y, you're asked to type in a number—this is the number of seconds that the Pi listens for a voice command and the default is 3.

The program then gives you a chance to set up the text-to-speech options. Be sure to turn up your volume before doing this. The program attempts to say something and asks whether you have heard it.

The default response of the system when responding to your keyword is 'Yes, sir?'. Choose y on the next prompt to change this, then type in your desired response, such as 'Yes, Mr Stark?' Press Return when you're done. The system plays back the response for you to confirm whether you're happy with the result.

The procedure is the same for the default message for when the system receives an unknown command. The default response is 'Received improper command', but you can change this to something less robotic if you prefer by typing y, then your chosen response – for example, 'I'm sorry, Dave, I'm afraid I can't do that.'

You are now offered the chance to set up the speech recognition options. This automatically checks whether you have a compatible microphone installed. *Voicecommand* next asks you if you want the Pi to check your audio threshold for you. Make sure there is no background noise, press y, then Return.

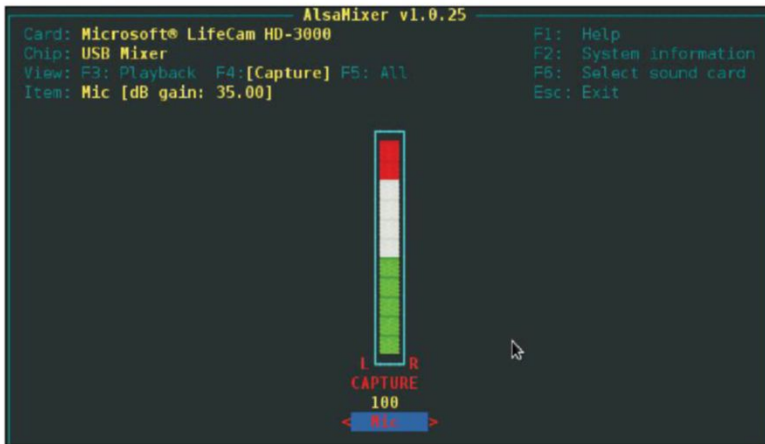
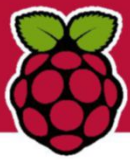
Quick tip

If, when testing text-to-speech, you hear the word 'fill', the filler flag should be set to 0. Press y to confirm you wish to change this.

```
Do you want to permanently change the default command duration of the sp
gnition (2 seconds)? (y/n)
n
Do you want to set up and check the text to speech options? (y/n)
y
First I'm going to say something and see if you hear it
Did you hear anything? (y/n)
y
If you heard the word FILL (or FILLER or FILLER FILL) at the beginning o
tence, the filler flag should be set to 0 or be blank
Do you want me to permanently set the filler flag to 0? (y/n)
y
The default response of the system after it finds the keyword is "Yes Si
Do you want to change the response? (y/n)
y
Type the phrase you want as the response:
Yes Ma'am
```

» **Simply type a new greeting and press Return. If you're not a Trekkie, you can also set the quiet flag, so the Pi doesn't respond verbally.**

» **Love your Pi more:** Subscribe and save at <http://bit.ly/LinuxFormat>



➤ Use the Up arrow to maximise the capture volume of your device (in this case, we're using a Microsoft USB webcam).

➤ Next, you'll be asked to speak a command to enable *Voicecommand* to check that it has the right audio device selected. The program automatically determines the right audio threshold for you, so type y to choose this.

Finally, the Pi asks you if you want to change the default keyword ('Pi') to activate voice commands. Type y, then enter your preferred keyword. Press Return when done.

You are then asked to speak your keyword to acclimatise the Pi to your speaking voice. If this seems correct, type y to complete the setup. Follow Step 6 of the tutorial walkthrough (see *right*) to run the *Voicecommand* software. Try to start out with a few simple commands. (See *Basic Voice Commands, on the previous page, for details*). Once you're comfortable with these, run the command `sudo killall voicecommand` to shut down the program and edit your configuration file if you wish.

Vexing voices

Voice recognition software is a work in progress and the Pi may not recognise everything you say. To improve your chances, be sure to stay near the USB microphone and speak slowly and clearly.

If you're still having trouble being understood, open Terminal on your Pi or connect via SSH and run the command `alsamixer` to open your sound settings. Press F4 to choose audio input, then press F6. Use the arrow keys to

select your USB device, then press Return. This controls the volume of your USB microphone. Use the up arrow to push it to maximum (100).

If your device isn't being detected at all, it may need more power than the Pi's USB ports can provide on their own. The best solution for this is to use a powered USB hub. The Pi Hut website sells a seven-port powered hub for just £11 (<https://thepihut.com/products/7-port-usb-hub-for-the-raspberry-pi>). This can also be useful if you want to use an external hard drive with your Pi.

Once the *Download* program is installed, if you experience an error connecting, bear in mind that access to the PirateBay website may be restricted where you are. In order to download files, you also need a BitTorrent client for the Pi, such as the program *Transmission*. You can install this by opening Terminal or connecting to your Pi over SSH and running the command `sudo apt-get install transmission`. Help with getting started and how to use the client is available from the *Transmission* website at <https://forum.transmissionbt.com>. Needless to say, you should only download files with the permission of the copyright holder.

If you choose to use *Google Voice Commands* or *Google TTS (Text to Speech)*, bear in mind that anything you say and any text files you submit are sent to Google's servers for translation. Google claims not to retain any of this data, but even if it is to be believed, any data transmitted over the internet can potentially be intercepted by a third party. Google does encrypt your connection to reduce the chance of this happening, however.

If you find you're happy with the voice command feature, you might prefer the software to start automatically each time you boot the Pi. If so, open Terminal on your Pi or connect via SSH and run the following command:

```
sudo nano /etc/rc.local
```

This opens the file that determines which processes start up when your Pi boots. By default, this script does nothing. Use your arrow keys to scroll to the bottom of the file and, just above the line reading `exit 0`, type the following:

```
sudo voicecommand -c
```

Press Ctrl+X, then Y, then Return to save your changes. Feel free to reboot the Pi at this stage to make sure it works. If you're unsure whether *Voicecommand* is running, open Terminal and run the command `ps -a` to show a list of running processes.

Quick tip

Choose a keyword that can't easily be confused with another word. For instance 'Pi' rhymes with 'hi' and 'bye', but it would be much harder for *Voicecommand* to mishear the word 'elephant... unless you pronounce 'elegant' in a strange way.

Voice command tweaks

Once your *Voicecommand* software is up and running, you can edit the configuration file to add new commands or modify existing ones.

Run the command `sudo nano /root/.commands.conf` to view the configuration file. As you'll see, most of the lines begin with a `#` symbol, which means the Pi ignores them. Delete the symbol to activate the line. If, for instance, you want to change the keyword that activates the voice recognition software from 'Pi' to 'Jessie', you would change the line from `#!keyword=pi` to `!keyword=jessie`. If you use the *Firefox* web browser instead of *Midori*, you may also want to change `~!Internet=midori &` to `~!Internet=firefox-esr &`.

The software can run any command. For instance, to open the desktop by saying the

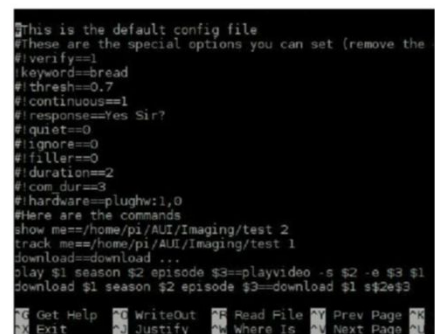
word 'desktop', add the following line to the end of the file: `desktop==home/pi/Desktop`. You can also launch programs as you would from the terminal, e.g. `notepad==leafpad`.

As you're talking to the Pi, you may want it to respond. Do this first by opening Terminal and installing the speech synthesis software, *Festival*, with the following command:

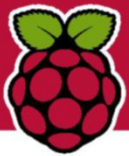
```
sudo apt-get install festival
```

The basic format to get the Raspberry Pi to talk is `echo "Your message here" | festival -tts`. You can also have the Pi read out system information. For example, if you wanted the Pi to tell you the date and time, you would add the following line to the config file:

```
time==echo "The time is" | festival -tts &&
date | festival -tts
```



➤ Remove the `#` at the start of a line to activate it. Change the value after `==` to alter your settings. Here the activation keyword has been changed to `bread`.



Set up voice commands on your Pi

```
nate@debian:~$ ssh pi@192.168.1.123
pi@192.168.1.123's password:

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: Wed Aug 17 14:37:26 2016 from 192.168.1.5
pi@raspberrypi:~$ git clone git://github.com/StevenHickson/PiAUISuite.git
Cloning into 'PiAUISuite'...
remote: Counting objects: 802, done.
remote: Total 802 (delta 0), reused 0 (delta 0), pack-reused 802
Receiving objects: 100% (802/802), 4.14 MiB | 1.30 MiB/s, done.
Resolving deltas: 100% (475/475), done.
Checking connectivity... done.
pi@raspberrypi:~$ cd PiAUISuite/Install
pi@raspberrypi:~/PiAUISuite/Install$
```

```
calibre Library      hpc.pdf             Release.key.2
decrypted            master.zip          Templates
decrypted2          Music               test1.pdf
Desktop             P-07:14:2016-13:20:41.jpg thunderbird
Documents           PiAUISuite         video.avi
Downloads           PiAUISuite-master  Videos
Dropbox             PiAUISuite.tar.gz  VirtualBox VMs
encrypted           Pictures

nate@debian:~$ cd PiAUISuite-master
nate@debian:~/PiAUISuite-master$ ls
DownloadController  Install  Makefile  PlayVideoScripts  TextCommand  Youtube
Imaging            LICENSE  Misc      README.md         VoiceCommand

nate@debian:~/PiAUISuite-master$ cd Install
bash: cd: Install: No such file or directory
nate@debian:~/PiAUISuite-master$ cd Install
nate@debian:~/PiAUISuite-master/Install$ sudo ./InstallAUISuite.sh
Installing AUI Suite by Steven Hickson
If you have issues, visit stevenhickson.blogspot.com or email help@stevenhickson.com
Install dependencies? y/n
These are necessary for any of the options, so you should probably press y unless
you absolutely know you have them already
```

1 Download installation files

Open Terminal on your Pi or connect via SSH and download the installation files by running the command `git clone git://github.com/StevenHickson/PiAUISuite.git`. When the download is complete run `cd PiAUISuite/Install` to open the installation directory.

2 Run the installer

Use the command `sudo ./InstallAUISuite.sh` to run the text installer. Choose `y` to install any dependencies and then work your way through the installation wizard. See the main article for more details about how to do this.

```
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: Wed Aug 17 14:42:43 2016 from 192.168.1.5
pi@raspberrypi:~$ sudo apt-get install libboost-regex1.49.0
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  libboost-regex1.49.0
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 0 B/390 kB of archives.
After this operation, 957 kB of additional disk space will be used.
Selecting previously unselected package libboost-regex1.49.0.
(Reading database ... 130761 files and directories currently installed.)
Preparing to unpack .../libboost-regex1.49.0-1.49.0-4+b2_armhf.deb ...
Unpacking libboost-regex1.49.0 (1.49.0-4+b2) ...
Setting up libboost-regex1.49.0 (1.49.0-4+b2) ...
Processing triggers for libc-bin (2.19-18+deb8u4) ...
```

```
nate@debian:~$ ssh pi@192.168.1.123
pi@192.168.1.123's password:

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: Wed Aug 17 14:43:19 2016 from 192.168.1.5
pi@raspberrypi:~$ sudo mv /home/pi/.commands.conf /root/.commands.conf
pi@raspberrypi:~$
```

3 Install Boost C++ library

If you encounter an error when trying to set up the *Voicecommand* program, run `sudo apt-get install libboost-regex1.49.0` to install the necessary software. Next, run `sudo ./InstallAUISuite.sh` to run the installer again.

4 Copy configuration files

Because you're going to be running the *Voicecommand* program as the root user, you need to use the command `sudo mv /home/pi/.commands.conf /root/.commands.conf` to copy across the necessary configuration files.

```
#!/filler==0
#!/duration==2
#!/com_dur==3
#!/hardware==plughw:1,0
#Here are the commands
show me==/home/pi/AUI/Imaging/test 2
track me==/home/pi/AUI/Imaging/test 1
download==download ...
play $1 season $2 episode $3==playvideo -s $2 -e $3 $1
download $1 season $2 episode $3==download $1 s$2e$3
desktop=home/pi/Desktop
play==playvideo -r -f ...
multiple==playvideo -r -m -c 5 ...
download==download ...
YouTube==youtube-search ...
Google==google ...

G Get Help  W WriteOut  R Read File  V Prev Page  X Cut Text
x Exit      J Justify    M Where Is  v Next Page  u UnCut Text
```

```
pi@raspberrypi: ~
File Edit View Search Terminal Help
nate@debian:~$ ssh pi@192.168.1.123
pi@192.168.1.123's password:

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: Wed Aug 17 14:46:37 2016 from 192.168.1.5
pi@raspberrypi:~$ sudo voicecommand -c
Opening config file...
running in continuous mode
keyword duration is 2 and duration is 3
```

5 Edit config file

Use the command `sudo nano /root/.commands.conf` to inspect your configuration file. You can now change your keyword or add extra commands. (See *Voice Command Tweaks*, left, for more details.)

6 Run Voicecommand

Use the command `sudo voicecommand -c` to start the software. It automatically listens out for your keyword. You can then give it further commands, such as 'YouTube fluffy kittens'. **LXF**

Issue 220
February 2017

Product code:
LXFDB0220

In the magazine

Make your increasingly clever home secure and connect to our pick of the best remote desktop clients. Plus, make your plots beautiful with D3.js, handle text in Python, web hosting with Drupal 8 and Linux on Dell devices.



LXF DVD highlights
Install the enterprise Linux distro openSUSE Leap 42.2 64-bit.

Issue 219
January 2017

Product code:
LXFDB0219

In the magazine

Our no nonsense guide to getting started with the greatest OS on the planet (the Martians are still using MacOS—losers). Plus our pick of the lightweight distros, build a faster Linux PC and inside Wayland.



LXF DVD highlights
Manjaro 16.10.2, Fedora 25, antix 16, Bodhi Linux 4 and more.

Issue 218
December 2016

Product code:
LXFDB0218

In the magazine

The ultimate guide to getting the ultimate Ubuntu and the best Chromebooks herded into a pile. Plus revive your old PC with a 32-bit distros, using Wireshark, learning about statistical learning and VPN.



LXF DVD highlights
BunsenLabs 2016.07.10, Ubuntu 16.10 32-bit & 64-bit and more.

Issue 217
November 2016

Product code:
LXFDB0217

In the magazine

Is it a bird, is it a plane? No, it's Super Pi! Explore how to build a slick Pi machine. Also this month, our secure chat app Roundup, recreate the Enigma machine, fun with stats, MythTV and build a CCTV sentry.



LXF DVD highlights
Android-x86 6.0, Elementary OS 0.4, Linux Mint 18 KDE and more.

Issue 216
October 2016

Product code:
LXFDB0216

In the magazine

Hackers! No, we say! Hunker down in LXF's digital bunker of privacy. Plus pick the best 3D printer for you, build an espresso machine or a dance mat and tackle Big Data, time syncing and bosh out some Bash.



LXF DVD highlights
Tails 2.5, Kali Linux 2016.1, AntiX 16 and SystemRescueCD.

Issue 215
September 2016

Product code:
LXFDB0215

In the magazine

We celebrate 25 years of the kernel and get excited about... accounting tools! If that wasn't exciting enough, how about the best distros? Plus: loads of tutorials, AI-brewed beer and a drone flying lesson.



LXF DVD highlights
Fedora 24, Neon 5.7.2, Voyager 16.04 and Ultimate Boot CD 5.3.

To order, visit myfavouritemagazines.co.uk

Select Computer from the all Magazines list and then select Linux Format.

Or call the back issues hotline on **0344 848 2852**
or **+44 344 848 2852** for overseas orders.

Quote the issue code shown above and have your credit or debit card details ready

Future

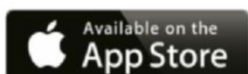
GET OUR DIGITAL EDITION!



SUBSCRIBE TODAY AND GET 2 FREE ISSUES*



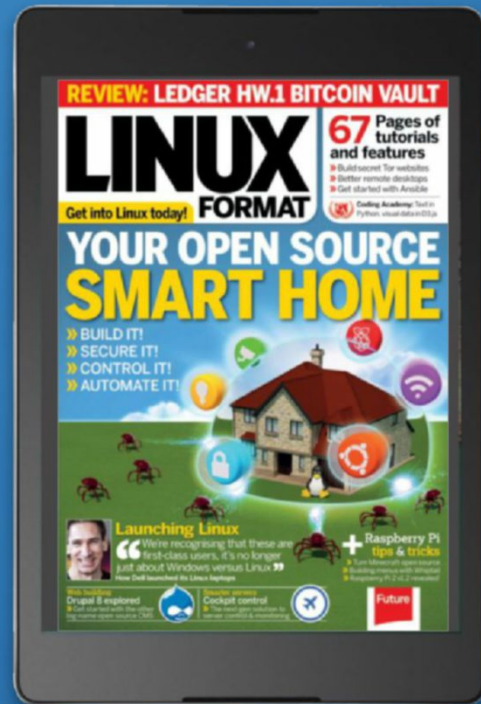
Available on your device now



*Free Trial not available on Zinio.

NOT FROM THE UK?

Don't wait for the latest issue to reach your local store – subscribe today and let *Linux Format* come straight to you.



"If you want to expand your knowledge, get more from your code and discover the latest technologies, *Linux Format* is your one-stop shop covering the best in FOSS, Raspberry Pi and more!"

Neil Mohr, Editor

TO SUBSCRIBE



Europe?

From €26 every 3 months



USA?

From \$29 every 3 months



Rest of the world

From \$38 every 3 months

IT'S EASY TO SUBSCRIBE...
myfavouritemagazines.co.uk/LINsubs17

CALL +44 344 848 2852

Lines open 8AM–7PM GMT weekdays, 10AM–2PM GMT Saturdays *

Savings compared to buying 13 full-priced issues. You will receive 13 issues in a year. You can write to us or call us to cancel your subscription within 14 days of purchase. Your subscription is for the minimum term specified and will expire at the end of the current term. Payment is non-refundable after the 14 day cancellation period unless exceptional circumstances apply. Your statutory rights are not affected. Prices correct at time of print and subject to change. * UK calls will cost the same as other standard fixed line numbers (starting 01 or 02) and are included as part of any inclusive or free minutes allowances (if offered by your phone tariff). For full terms and conditions please visit bit.ly/magtandc. Expiry date in the terms: 31 March 2017

» **Terminal** Learn the useful core commands for administering a server

Admin: Core commands

20 terminal commands that all Linux web server admins should know.



Our expert

Adam Oxford
runs South African
tech news site
www.htxt.co.za.
He learned many
of these lessons
the hard way.

Are you an 'accidental admin'? Someone who realised, too late, that they were responsible for the workings of a Linux server and – because something has gone wrong – finds themselves lost in a world of terminals and command lines that make little sense to normal humans?

What is SSH, you may be asking yourself. Do those letters after 'tar' actually mean anything real? How do I apply security patches to my server? Don't worry, you're not alone. And to help you out, we've put together this quick guide with essential Linux commands that every accidental admin should know.

Becoming an accidental admin

While we'd argue that they should, not everyone who starts using Linux as an operating system does so through choice. We suspect that most people's first interaction with Linux happens somewhat unwittingly. You click a button on your ISP's account page to set up a personal or business web server – for a website, email address or online application – and suddenly you're a Linux admin. Even though you don't know it yet.

When you're starting out with your web server, things are usually straightforward. Nearly all hosting providers will give you a web interface such as Cpanel or Plesk to manage your server. These are powerful pieces of software that give you quick and easy access to logs, mail services and one-click installations of popular applications such as Wordpress or forums. But the first time you have to do something that isn't

straightforward to do through the graphical control panel, you're suddenly out of the world of icons and explanatory tooltips and into the world of the text-only Terminal.

To make things worse, for a lot of people the first time they have to deal with the Terminal is when something has gone wrong and can't be fixed through the control panel. Or perhaps you've just read that there's a major security flaw sweeping the web and all Linux servers *must* be updated at once (it happens – search for 'Heartbleed' to find out more). Suddenly you realise that your nice control panel hasn't actually been updating your server's operating system with security patches and your small personal blog may well be part of a massive international botnet used to launch DDOS attacks against others. Not only are you a stranger in a strange land, you're probably trying to recover or fix something that was really important to you, but which you never gave much thought to while it was being hosted for a couple of pounds a month and seemed hassle-free.

You are an 'accidental admin'. Someone who is responsible for keeping a Linux webserver running and secure—but you didn't even realise it. You thought all that was included in your couple of pounds a month you pay to your ISP – and only found out it's not when it was too late.

Since most web servers are running Ubuntu, this guide is based on that particular distribution. And all the commands here are just as applicable to a Linux desktop as they are to a web server, of course.

1 sudo

The most fundamental thing to know about Linux's approach to administration is that there are two types of accounts that can be logged in: a regular user or an administrator (aka 'superuser'). Regular users aren't allowed to make changes to files or directories that they don't own—and in particular this applies to the core operating system files which are owned by an admin called 'root'.

Root or admin privileges can be temporarily granted to a regular user by typing **sudo** in front of any Linux command. So to edit the configuration file that controls which disks are mounted using the text editor, *nano*, you might type **sudo nano /etc/fstab** (we really don't recommend this unless you

» Can't remember that really clever thing you did last week? History is your friend.

```
670 df
671 nan df
672 df -h
673 top
674 w
675 nano
676 dir
677 history
678 diff -r /media/studiopc/theRaptorRaid/Pics /media/sdb1/Pics
679 sudo su
680 ssh root@crazy@ball.htxt.co.za
681 ssh root@crazy@ball.htxt.co.za
682 ssh root@htxt.co.za
683 ssh root@41.79.76.130
684 ifconfig
685 ls | less
686 utt
687 ls | less
688 apt-get update
689 sudo apt-get update
690 history
studiopc@studiopc:~$
```


Connecting to the server

As an accidental admin, your first challenge is going to be connecting to your server in the first place. In your web control panel, you might see an option to open a Terminal or console in your web browser, but this tends to be quite a laggy way of doing things.

It's better to open up a Terminal window on your own machine (if you're running Ubuntu just press Alt+Ctrl+t, if you're on Windows you'll need an application like *PuTTY*). Now, at your command prompt, type `ssh username@`

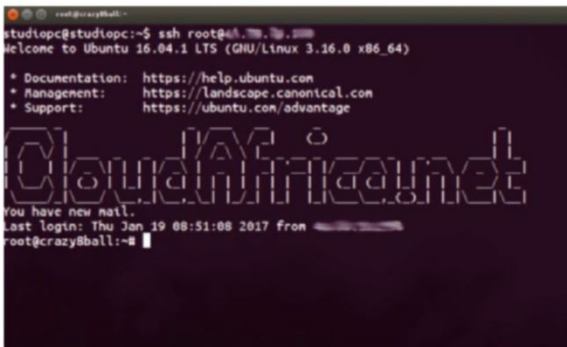
`yourserver.com` (or you can replace `yourserver.com` with an IP address).

The `ssh` command will open a secure shell on the target machine with the specified username. You should get a password prompt before the connection is allowed and you will end up in a text interface that starts in the **home** folder of the username.

If you're going to be connecting regularly, there's an even more secure way of using `ssh` and that's to bypass the password prompt all

together and use encrypted keys for access instead. To follow this approach, you'll need to create a public/private SSH keypair on your machine (for example, Ubuntu users can type something like `ssh-keygen -t rsa -b 4096 -C "your_email@example.com"`) and copy the public part of the key into the `.ssh` folder on the target server.

You will find some full instructions for doing this here: <https://help.github.com/articles/generating-an-ssh-key>.



» Even if someone copies your key, they'll still need a password to unlock it.

know what you're doing). After entering `sudo`, you'll be asked for your user password. On a desktop PC, this is the same one that you use to log in. If you're logging into your own webserver, however, there's a good chance that you'll already be the root user and won't need a password to make important changes.

If you can't execute `sudo` commands, your web host has restricted your level of access and it probably can't be changed. User accounts can be part of 'groups' in Linux and only members of the sudoers groups can use the `sudo` command to temporarily grant themselves admin privileges.

2 su

While `sudo` gives you great power, it still has limitations. Most of all, if you've got a whole bunch of commands to enter, you don't want to have to type it out at the start of every single line [at least the password has a 5 minute timeout-Ed]. This is where `su` comes in, which will give you superuser powers until you close the terminal window. Type `sudo su` followed by your password, and you'll see the prompt change from `yourname@yourserver` to `root@yourserver`. You might think `su` stands for superuser, but it's actually a command to change to any user on the system and if it's used without an account name after it, `su` assumes you want to be root. However, using `su myname` will switch you back to your original, non-super, login.

3 ifconfig

Since you're troubleshooting a web server, it's probably a good idea to get as many details about its actual connection

as possible noted down. The `ifconfig` command can be run without `sudo` privileges and tells you details about every live network connection, physical or virtual. Often this is just for checking your IP address, which it reports under the name of the adaptor, but it's also useful to see if you're connected to a VPN or not. If a connection is described as `eth0`, for example, it's an Ethernet cable meanwhile `tun0` is a VPN tunnel.

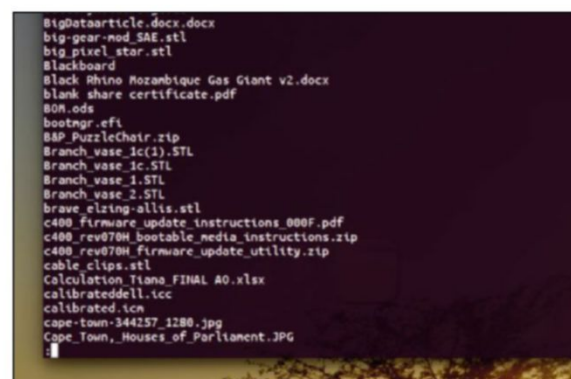
4 chown

There's tons more you can learn about `chmod` and we strongly recommend that you do, but it has a sister command that's even more powerful. While `chmod` dictates what users who aren't the owner of a file can do, the `chown` command changes the file owner and group that it belongs to completely. Again, you'll probably need to put `sudo` in front of anything you `chown`, but the syntax is again simple. An example might be `chown myname:mygroup filename.file`.

5 service restart

No, we're not telling you to 'try turning it off and on again', but sometimes it's a good place to start (and sometimes it's essential to load changes into memory). It's possible you might be used to start and stop background processes on a Windows desktop through the graphical System Monitor or Task Manager in Windows. However, in the command line Terminal to a server it's a little more tricky, but not by much.

Confusingly, because many Linux distributions have changed the way they manage startup services (by switching to `systemd`) there's two ways of doing this. The old way, which still works a lot of the time, is to just type `service myservice restart`, preceded with `sudo`, when it's necessary. The new, correct, way is a little more verbose: `systemctl restart`.



Quick tip

If you're changing names, permissions or ownership most commands have a `-R` or `-r` option, which stands for 'recursive'. Essentially, this changes the attributes of all files inside a folder, rather than just the folder itself.

» Unless you can read 1,000 lines a second, you'll need to use `ls | less` to explore folders.

» Enhance your Terminal-fu Subscribe now at <http://bit.ly/LinuxFormat>

» **myservice.service**. So if you want to restart Apache, for example, the core software which turns a mere computer into a web server, it would be **sudo systemctl restart apache2.service**.

6 ls

The key to understanding the console is all in the path (see *Path To box, below*), which tells you whereabouts you are in the folder structure at any given time. But how do you know what else is in your current location? Easy: you use **ls**. The **ls** command lists all the files within the folder that you're currently browsing. If there's a lot of files to list, use **ls | less** to pause at the end of each page of filenames.

7 cat

A command you'll often see if you're following instructions you've found online – and aren't always sure what you're doing – **cat** is short for concatenate and is used to combine files together. In its simplest form it can be used to take file1.txt and file2.txt and turn them into file3.txt, but it can also be combined with other commands to create a new file based on searching for patterns or words in the original.

Quite often you'll see **cat** used simply to explore a single file – if you don't specify an output filename, **cat** just writes what it finds to the screen. So online walkthroughs often use **cat** as a way of searching for text within a file and displaying the results in the terminal. This is because **cat** is non-destructive—it's very hard to accidentally use **cat** to change the original file where other commands might do.

8 find

A useful and under used command, the **find** command is pretty self-explanatory. It can be used to find stuff. Typing it by itself is much like **ls**, except that it lists all of the files within

sub-directories of your current location as well as those in your current directory. You can use it to search for filenames using the format **find -name "filename.txt"**. By inserting a path before the **-name** option, you can point it at specific starting folders to speed things up. By changing the **-name** option you can search by days since last accessed (**-atime**) or more.

9 df

Maybe your server problems are to do with disk space? Type **df** and you'll get a full breakdown of the size and usage of every volume currently mounted on your system. By default it'll give you big numbers in bytes, but if you run **df -h** (which stands for 'human readable' the volume sizes will be reported in megabytes, gigabytes or whatever is appropriate.

10 apt-get update && upgrade

Probably the single most important command to know and fear. We all know that to keep a computer system secure you need to keep it updated, but if you've got control of a Linux box the chances are that it isn't doing that automatically.

A simple **sudo apt-get update** will order your system to check for the latest versions of any applications it's running, and **sudo apt-get upgrade** will download and install them. For the most part these are safe commands to use and should be run regularly—but occasionally updating one piece of software can break another, so back-up first...

11 grep

As computer commands go there are few more fantastically named for the newcomer than the **grep** [it's a real verb!–Ed] command. How on earth are you ever going to master this Linux stuff if it just makes words up? But **grep** is a great utility for looking for patterns within files. Want to find every line that talks about cheddar in a book about cheeses? **grep "cheddar" bookofcheese.txt** will do it for you. Even better you can use it to search within multiple files using wildcards. So **grep "cheddar" *.txt** will find every text file in which cheddar is reference. So now you grok **grep**, right?

12 top

When you're working in a graphical user interface such as a Linux desktop environment or Windows desktop, there's always an application like System Monitor or Task Manager which will call up a list of running applications and give you details about how many CPU cycles, memory or storage they're using. It's a vital troubleshooting tool if you have a program that's misbehaving and you don't know what it is.

In a similar way, you can bring up a table of running applications in the Linux Terminal that does the same thing by typing **top**.

» Nano isn't the only terminal text editor, but it's the easiest to use.

Path to

When you open a Terminal window within Linux, it can be a bit disorientating. But the words that sit in front of the flashing cursor will tell you where you are.

The first word is the name of the user you're logged in on, and it's followed by an '@' sign. The second word is the hostname of the

machine you're logged into. If you open up a Terminal on your desktop, usually the username and hostname are the same. So you'll see 'myname@myname'. When you log into a remote server, though, they'll be very different.

This information is followed by a colon which is followed by the path to the directory you're in,

followed by a dollar sign. When you first open a Terminal, it will usually print yourname@yourname:~\$. The tilde '~' indicates you're in the **home** folder for your username. If the dollar sign is replaced with a '#', you're using the machine as a root user. See **cd** for moving around and watch how the path changes as you do.

» **Never miss another issue** Subscribe to the #1 source for Linux on page 26.

20. chmod

User permissions are one of the most important parts of Linux security to understand. Every file has a set of permissions which defines who can see a file; who can read and write to a file; and who can execute a file as a program.

A file which can be seen by web visitors, but can only be changed by a specific user, is just about as basic as it gets when it comes to locking down a server. The problem is that some files need to be changeable and some don't—think of a Wordpress installation for a blog. You want

Wordpress to be able to write some files so it can update them, but there's also a lot of files you don't want it to be able to change—and you really don't want to give it power to execute code unless you have to. The flipside is that problems with web servers can be traced back to incorrect file permissions, when an app needs to be able to modify a file but has been locked out by default.

Your friend in this area is `chmod`. It changes permissions for which users and groups can read, write or execute files. It's usually followed

by three digits to indicate what the owner, members of its group and everyone else can do. Each digit from 0-7, where 7 allows for read, write and execute and 1 is execute only. If your user 'owns' the file in question, the syntax is simple.

`chmod 777 filename`, for example, will give all users the ability to read and write to a file. It's good practice not to leave files in this state on a webserver—for obvious reasons. If you don't own the file, you'll need to add `sudo` to the front of that command.

Like a lot of command line utilities, it's not immediately obvious how you can close **top** once you're finished with it without closing the terminal window itself—the almost universal command to get back to a prompt is Ctrl+c.

13 kill, killall

Using **top** you can figure out which application is using all your CPU cycles, but how do you stop it without a right-click > End process menu? You use the command **kill** followed by the process name. If you want to be sure and kill every process with a name that contains that application name, you use **killall** . So **kill firefox** will close down a web browser on a Linux desktop.

14 W

From the weirdness of `grep` to the elegance of the `w` command, a whole command in a single letter. If you think another user is logged into your system, this is an important command to know. You can use `w` to list all currently active users, although don't rely on it too much as it's not hard for a hacker to be hidden.

15 passwd

You must use `passwd` with extreme care. Ultra extreme care. Because the next word you write after it will become your login password, so if you type it incorrectly or forget it, you're going to find yourself in serious trouble.

You can only change your own user's password by default, but if you grant yourself sudo powers you can change any user's credentials by including their username after the password itself. Typing `sudo passwd`, meanwhile, will change the password for root.

Check out the manual (`man passwd`) page for some useful options to expire passwords after a certain period of time and so on.

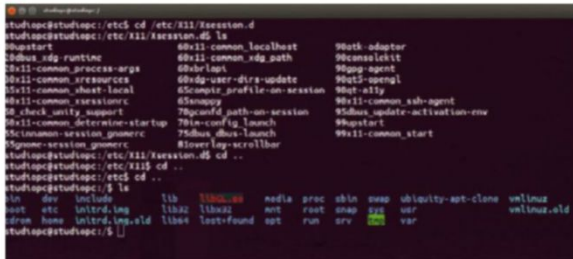
16 cd

If you have a graphical interface and file browser, it's pretty easy to move to new locations on your hard drive just by clicking on them. In the Terminal, we know where we are because of the path (see *Path To box*, left), and switch location using `cd` which stands for 'change directory'.

The `cd` command is mainly used in three ways:

1 cd foldername This will move you to that folder, provided it exists within the folder you're currently browsing (use **ls** if you're not sure).

2 cd ~/path/to/folder This will take you to a specific location ~ within your **home** folder (the ~ character tells **cd** to start looking in your **home** folder). Starting with a / will tell **cd** to start the path at the **root** folder of your hard drive.



3 cd .. This final useful command simply takes you up one level in the folder structure.

17 mv & rm & cp

When you get the hang of it, using a terminal as a file manager becomes pretty simple and quite a joyful experience. As well as `cd`, the three fundamental commands are `mv`, `rm` and `cp`. The `mv` command is used to move a file from one location to another, `rm` is used to remove or delete a file and `cp` will copy files and folders.

Just as with `cd`, you can either enter a filename to operate on a file in the directory you're working in or a full path starting from the root of the drive with `~`. For `mv` the syntax is `mv ~/location1/file1.file ~/location2/location`.

The big thing to remember is that in the Terminal there's no undo or undelete function: if you **rm** a file and it's gone forever (or at least will require very specialist skills to retrieve) and in a similar fashion, if you **mv** or **cp** a file you'd better make a note of where it went.

18 nano

It might seem odd, if you've spent your life in graphical applications and utilities, but complex programs run in the text terminal, too. There are several text editors which normally come as part of the whole package, notably *nano* and *vi*. You can open a blank document by typing `nano .`, or you can edit an existing one by typing `nano ~path/to/text.txt` (and do the same with *vi*). Some of the terminology may seem odd, though: To write out (Ctrl+o) means save, for example and so on.

19 history

And finally, if you've been copying and pasting commands from the web all day, you might want to check up on what you've actually done. You can use **history** to give you a list of all the terminal commands entered going back a long, long way. Execute specific numbered commands with **!, you can go back through recent commands just by using the up and down arrows (and re-issue them by tapping Enter), or search for commands by pressing Ctrl+r, **LXF****

➤ **Keep an eye on the directory path in front of the command line to figure out where you are.**



One command that's invaluable is **man** which is short for 'manual'. This will open up the help file for any other command. So if you want to know all the options for the **ls** command, simply type **man ls** and see what comes up.

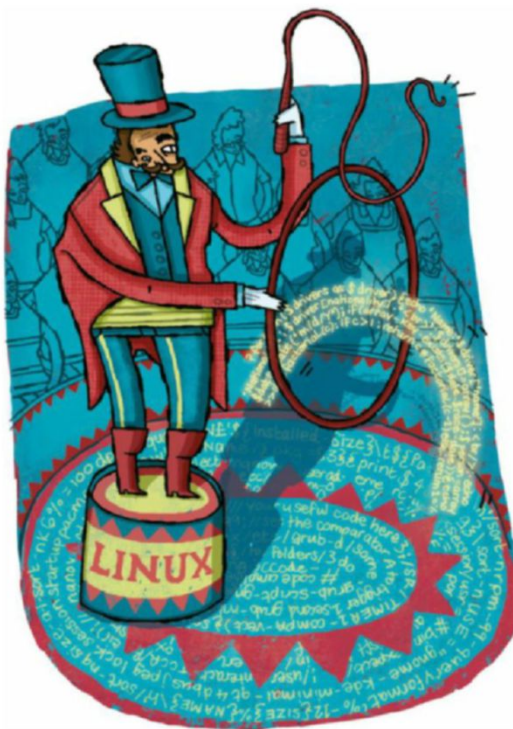
Ubuntu: Linux on your laptop

Adam Oxford picks up on the pain points that can bring Linux laptops low when you decide to liberate your portable devices.



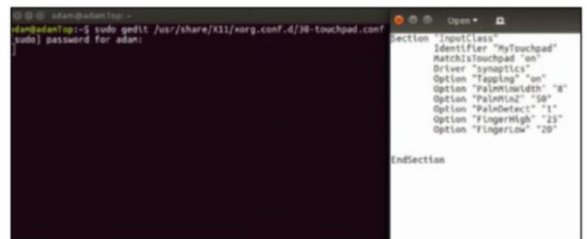
Our expert

Adam Oxford has been writing about tech for 20-odd years and using Linux exclusively for about half of that. His current business, htxt.co.za, is based in the spiritual home of Ubuntu, Johannesburg.



Quick tip

If you are struggling with battery life, *PowerTOP* is a useful tool for getting a list of which applications are using the most of your precious charge. Open up a terminal and type `sudo apt-get install powertop` and enter your password when prompted. Finally, type `powertop` to check your apps for power use.



» Learning to master X11's input devices will help with quirky hardware components in your laptop.

(distro) on a laptop is as quick and painless as it is on a desktop.

In this guide, we're going to show you how to deal with the few foibles you might find, and how to get the absolute most out of putting Linux on a portable PC. For the purposes of this walkthrough we're going to be focussing on the Ubuntu distro of the Linux desktop, because it's stable, scales well for modern laptops with high definition screens and is – arguably – the best Linux distro to use with a touchscreen. Personally, we'd prefer the Unity desktop to GNOME or KDE for touch and smaller screens in general—but the instructions are similar for any major distro.

Getting started

Installing Ubuntu on a Windows machine couldn't be simpler—there's plenty of guides for downloading the operating system and creating a bootable USB key to install from, so we won't go over that here. The process is more or less the same as installing Ubuntu on a desktop, but there are a couple of potential issues that may trip you up.

One thing we would advise, however, is to use the latest Long Term Support (LTS) distribution from Ubuntu. At the moment this is version 16.04, not the most recent release, 16.10. The reason for this is that newer and more experimental features introduced in none-LTS versions seem more likely to not work on notebooks.

With your LTS installation media in hand, you'll need to figure out where you're going to install it. Laptop storage is much more sacred than that of a desktop—especially if you have a high-end notebook with a solid-state drive. Finding a laptop you can open up to add hard drive space to is increasingly tough and as disk drives get faster and more power efficient they're also getting smaller.

That's not such an issue if you're going to delete the Windows partition of your machine completely, but if you want to set up a dual-boot system with an existing Windows partition, we'd only advise trying this if your entire hard drive

Laptops can be tricky things to get your favourite brand of Linux distribution (distro) working on. Despite the fact there are fewer ways for an owner to physically customise a laptop – you can't upgrade the graphics card like you can on a desktop – there can be more variation in terms of component combinations even within the same model number than there is for a desktop PC.

In particular, hard drives and wireless chips tend to be slightly less well-supported than desktop versions because the technology tends to evolve faster in mobile than on desktop and as more and more laptops are coming with ultra-high definition screens that can cause problems for older software that doesn't scale to high dots per inch (DPI) resolutions like 4K well.

Linux takes slightly longer to adapt to product upgrades than Windows does as manufacturers of Wi-Fi chips built into notebooks rarely release open source drivers, so it can take a little time for the Linux kernel to catch up with a small modification to a product line—but it always does and it's getting quicker at doing so. While some problems still exist, more often than not installing a popular Linux distribution

The Swiss Army knife of power saving

There are a lot of settings you can configure in Linux to help with battery life: from ordering the CPU to slow down or fiddling with disk settings to stop hard drives whirring into action unnecessarily. Over the years, distros such as Ubuntu has become pretty good at managing hardware in the most efficient way possible, but if you do want to try and improve battery life, you could do worse than install *TLP*. It's not as

effective as it used to be, but on low-end hardware it may yield good results.

TLP is designed to give you full control over settings that affect power consumption: from turning off SATA interfaces completely to reducing power to the Wi-Fi radio. It's complex and you should be careful while using it, but it comes with a default configuration that will apply the most effective settings safely.

Install *TLP* by opening up a terminal and adding Linrunner's repos by typing this:

```
sudo apt-add-repository ppa:linrunner/tlp
```

Now enter your password, and you'll need to type `sudo apt-get update` followed by:

```
sudo apt-get install tlp
```

Now type `sudo tlp start` to get things started.

Find out more here: <http://linrunner.de/en/tlp/tlp.html>.

space is 256GB or greater. When installing Ubuntu, we'd recommend that you shrink down the size of the Windows partition to about 100GB and create two new ones.

The first partition should be about 100GB and formatted as an NTFS drive, so it can be read by Ubuntu and Windows—this gives you a space to store files that you can work on from either OS. The second new partition will be what's left – around 50GB – and just for Ubuntu. The reason for this is that Windows is far more likely to grow over time than Ubuntu: after several years of upgrades and installs, the operating system on our Linux box is still only about 12GB (not including the **Home** folder). On our six-month-old Windows box, meanwhile, Microsoft's operating system takes up 27.7GB and for some inexplicable reason has held on to another 15GB in a **Windows.old** folder because there was a point release a few months ago.

So score one for Ubuntu. If you've got limited space on a laptop drive you can go for much longer before engaging in extreme housekeeping and freeing up disk space with Linux.

The second thing to be aware of is that quite a few new laptops now come with a NVMe compatible SSD drive, as opposed to the older mSATA standard. This is especially true if you're buying a 'premium' laptop that's superthin and light. In itself, NVMe isn't a problem, as Ubuntu fully supports installation onto this type of drive. But if you've bought a Windows laptop that you want to replace the OS or dual boot with, there may be a problem.

Several of these machines ship with a single NVMe drive configured in the BIOS as a RAID array. RAID is usually only for configuring multiple drives to work together, but in this case it's activated in order to allow some extra features under Windows. If you try to install Ubuntu directly onto an NVMe drive in RAID mode, you'll probably find that the drive won't be visible to the installer software. You'll have to go into the BIOS and change the interface type to AHCI rather than RAID.

This is easy enough, but if you want to keep your Windows installation intact, before you change the BIOS setting you'll need to boot into Windows, open the command prompt as an administrator and enter `bcdedit /set {current} safeboot minimal`. Reboot, change the BIOS setting, then go back to your admin prompt and enter `bcdedit /deletevalue {current} safeboot`. Now you can install Ubuntu as normal.

Wi-Fi worries

The single-most common issue people encounter when installing Ubuntu (or most other Linux distros) to their laptop is that Wi-Fi may not work straight away. As explained above, this is because Wi-Fi chips, in particular, are changed and

upgraded faster than the Linux kernel maintainers can keep up, so you may need to add some drivers to your laptop.

We don't have room to provide solutions for every laptop out there, but in general the best advice is to do a web search for the make and model of your particular laptop with 'install Ubuntu' before you begin, to see if there's any common problems and solutions already found so you can prepare the fix before you begin. Most of the time, the solution will simply involve updating Ubuntu itself—but you'll need internet access to do this.

So if your laptop doesn't come with an Ethernet port built-in, either make sure you have a USB to Ethernet adaptor handy or an older USB Wi-Fi dongle that you've tested works with Linux support in another machine.

If anything else isn't working after an installation, a handy place to start troubleshooting is to open a terminal window (press Alt+Ctrl+T by default) and enter `lspci -nn`. This handy command will list all the hardware currently installed along with model number and manufacturer.

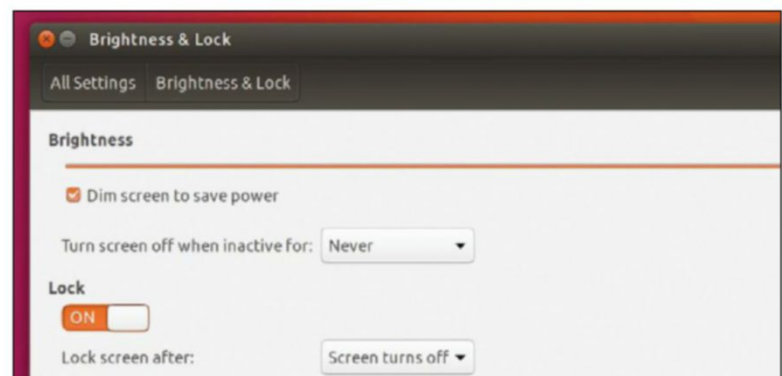
Pointer problems

Your laptop almost certainly has a trackpad on it. Trackpad support is good on Ubuntu, but as with Wi-Fi chips there's a lot of variety in the actual hardware under your thumbs which can create a few quirky issues.

Ubuntu does support multitouch gestures like tapping with two fingers to simulate a right-mouse click or three fingers for a middle click, or using two fingers to scroll vertically and horizontally. It doesn't however, support other common gestures, like pinch-zoom. Some people have had success getting these to work by using an application called *Toucheegg*, but it's luck of the draw whether or not its compatible with your laptop.

Quick tip

If your primary work machine is your notebook, you probably don't want it turning off every few minutes while you wander off to grab a glass of water. In System settings > Power you'll find options for customising when your machine goes into low power mode. Setting it to 'Don't suspend' while plugged in will reduce the number of times you have to wake it every day.



» Reducing screen brightness is the single best thing for improving battery life.

» **Want even more Linux tips?** Grab a bookazine at <http://bit.ly/LXFspecial>

» Ubuntu supports other trackpad features like 'disable while typing' (which stops palm taps on the trackpad moving the cursor around while you're writing in a document). The problem is that due to some quirks of the display manager, these aren't always available in Settings > Mouse screen. You can see a full list of settings for your trackpad by going to a terminal and entering `synclient -l`.

Here you'll see settings such as 'Tapping' 'On' and 'FingerLow' '20' which determine how your trackpad behaves. There's a full list of settings here: https://wiki.archlinux.org/index.php/Touchpad_Synaptics.

You can create a configuration file to change the way your trackpad works. If you want to turn on palm detection/disable while typing, for example, go to the terminal and type `sudo gedit /usr/share/X11/xorg.conf.d/30-touchpad.conf` followed by your password at the prompt.

In the empty document that appears, type the following:

```
Section "InputClass"
    Identifier "MyTouchpad"
    MatchIsTouchpad "On"
    Driver "synaptics"
    Option "PalmDetect" "1"
EndSection
```

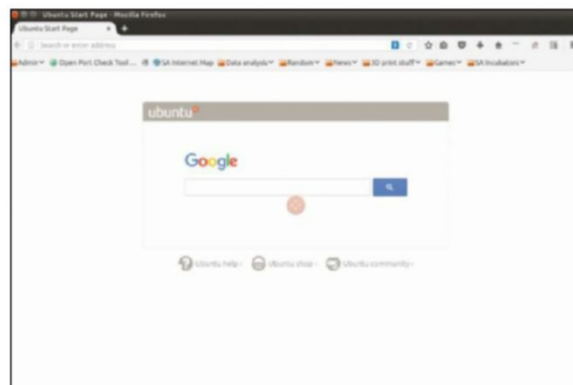
And save and restart. You can add extra lines before the `EndSection` comment to enable other features or fine-tune your settings.

If you find that changing the Synaptics settings isn't having any effect, we have come across a number of laptops which detect a single trackpad as both a USB and a PS/2 mouse. This creates an odd conflict where the mouse that your laptop thinks you're using isn't actually the virtual mouse that the settings are prepared for. Open a terminal and type `xinput list`. This will give you a list of all the input devices connected to your laptop. Assuming that you don't have a separate USB mouse connected, there should only be one entry for a touchpad (you may see your touchscreen listed here too).

If you see two touchpads, type:

```
sudo gedit /etc/modprobe.d/blacklist.conf
```

and enter your password when prompted. You will need to add a line at the bottom of that file that says `blacklist psmouse` and save the configuration file. Now type `sudo depmod -a` and reboot. Next time you run `xinput list`, you should only see one touchpad and all your mouse settings should be active.



» Got a touch screen? Ubuntu has a few useful built-in gestures for slinging windows around and maximising with a three-fingered flick.

The other thing that might crop up if you're using a newish laptop is that you might have a touchscreen. Generally speaking, we've found that these are supported well by Ubuntu these days and it shouldn't give you any problems. We do recommend using *Chromium* rather than *Firefox* as your web browser of choice however, because it supports gestures, such as pinch-zoom and finger scrolling much better. To try *Chromium* – which is more or less identical to Google's *Chrome*, but open source—open up the *Ubuntu Software* application and search for it there.

High DPI

Sticking with the subject of screens, and if you're lucky enough to have a high-definition screen on your machine, don't fret. Machines with resolutions higher than 1,920x1,080 are increasingly common, and they look beautiful with crisp fonts and sharp images. Fortunately, Ubuntu supports them all, although it requires a small tweak after installing.

When you first boot into your new operating system, you might be terrified by the fact that all the icons, task bar and text are tiny. Fixing this is simple: squint a bit while you open up the System Settings (by typing `settings`) into the launcher, and click on 'Displays'.

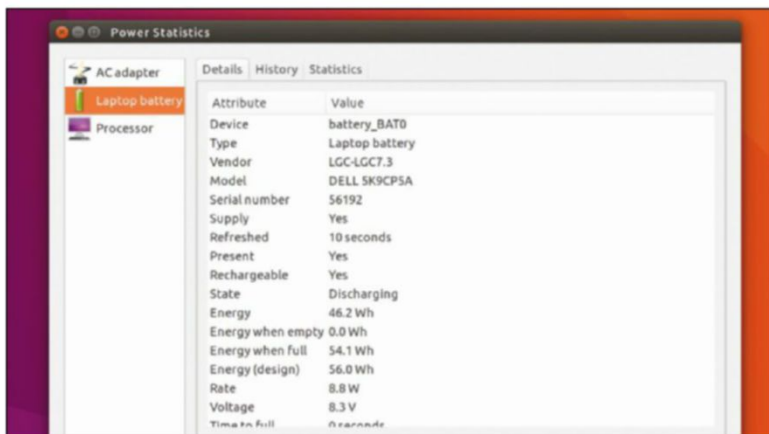
The slider at the bottom of this screen is called 'Scale for menu and Title Bars'. Setting this to 2. This will draw icons and text at twice their normal size—usually sorts this out. Most applications will scale automatically to the new desktop ratio too—notable exceptions include the GIMP graphics program, which works best if you install a high-resolution theme specific to that application.

The other place a high DPI (dots per inch) screen might catch you out is if you're using a second monitor with your laptop. Second screens are an almost essential item these days for those who work on a notebook and increase productivity or make gaming a joy when you're sat at your desk when you should be working. The last thing you want is to be hunched over a laptop's tiny display longer than you have to be.

Ubuntu and Linux in general is very good with dealing with multi-monitor setups. In System Settings > Displays you'll find all the options that you need to set resolutions and relative positions for using more than one monitor at a time. If one of your screens is much higher resolution than the other, however, you might run into issues as Linux isn't quite

Quick tip

From a terminal, type `sudo apt-get install indicator-cpufreq`. This will put a small icon in your top bar that enables you to change the CPU governor from 'Powersave' to 'Performance'. It doesn't have a huge effect on modern laptops, but may help a little with low-end ones. If you use this tool on older versions of Ubuntu than 16.04, you can directly set the CPU speed from the indicator.



» This handy dialogue tells you everything you need to know about your battery.

» **Never miss another issue** Subscribe to the #1 source for Linux on page 26.

as good at handling different scaling factors as other operating systems yet.

The problem for fancy Retina-type high definition displays is that while you can set resolutions independently for each screen, you can only have one scaling factor for the desktop furniture. If you have a laptop that has a 3,200x1,800 built-in display that's hooked up to a 1,920x1,080 monitor, you can either have things looking normal on the laptop and huge on the big screen (scaling factor 2) or normal on the big screen and tiny on the laptop (scaling factor 1).

The workaround that we're most comfortable with is to connect the second screen and then set the laptop monitor to 1,920x1,080 resolution as well and move the scaling factor to 1. It's not perfect – you lose the benefit of the high DPI resolution while you're at your desk – but both screens should look 'normal'.

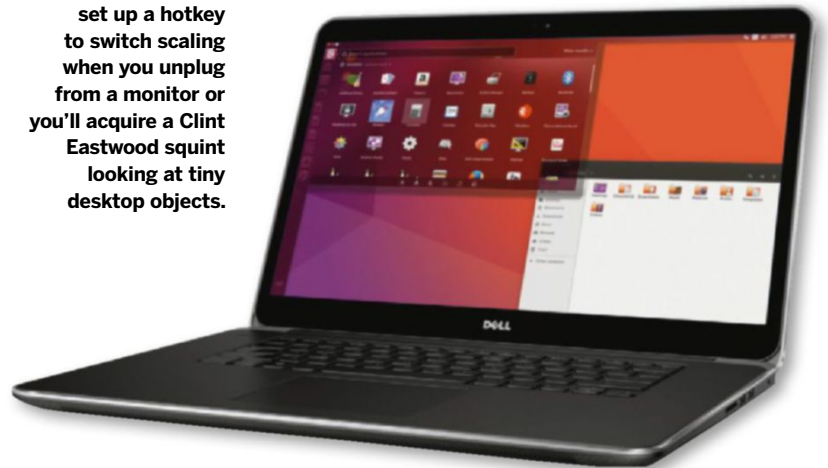
Ubuntu will remember the resolution settings too – when you unplug the second screen the built-in one will revert back to 3,200x1,800 (for example). When you plug it back in, the resolution will reduce. What won't change automatically, however, is the scaling factor. When you unplug your screen then objects and text will seem tiny until you change the setting back to 2. You can manually change this in the settings each time, but this gets boring fast. Better to set a hotkey to switch between scaling factor 1 and 2. This is a little tricky to do (*see the Quick Scaling walkthrough, below*).

Power saver

If you're like us, you'll want to squeeze every last second out of your laptop's battery during a long day at work, and the good news is that there's not much difference, if any, between the length of time a laptop with Ubuntu installed lasts and the length of time the same portable will survive with Windows running on it. Ubuntu has become really good at power management over the last few years.

The single best thing you can do to save power is simple: turn down your screen brightness when your machine isn't charging. Most of the power used by a contemporary laptop goes into lighting up all those pixels, so turning it down can easily double the time a charged power pack will last. Beyond that, there's a few other simple things to try.

» You'll need to set up a hotkey to switch scaling when you unplug from a monitor or you'll acquire a Clint Eastwood squint looking at tiny desktop objects.



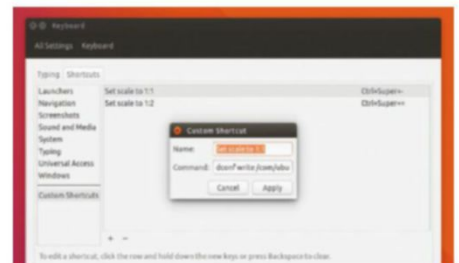
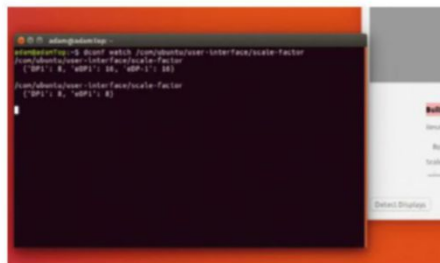
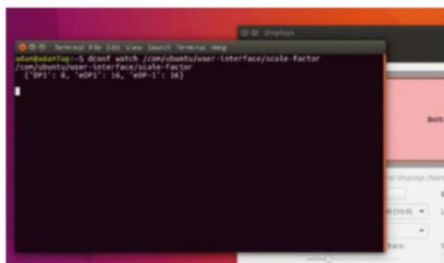
First of all, if you use *Chrome* or *Chromium* as your browser, make sure you go into the settings and turn off the option that lets the web browser run in the background. We've found that Google's browser can eat tons of resources if you let it, so stay in control. Similarly, not leaving your mail client running and constantly pinging servers helps too.

If you want to see more details about your current battery use and anticipated charge time, click on the battery icon in your notification area and then click on the first line in the drop-down menu with the icon in it. This screen will chart the discharge time over the last 24 hours and give you loads of information about the state of your battery.

The 'Rate' is the current power draw, which could be anywhere from 5W to 15W depending on how efficient your laptop's processor is and how well your system is set up. 'Energy when full' tells you how much charge your battery can hold. If this is significantly lower than the 'Energy (design)' – which is how much the battery is supposed to hold – it's time to get a new battery.

But hopefully it won't come to that. If you keep an eye on your portable and take care of your notebook following our instructions and you'll have many happy years of Linux laptopting ahead of you. **LXF**

Quick scaling



1 Watch your settings

Got a high DPI screen and tired of having to open up System settings to change the display scaling whenever you plug in an external monitor? Here's how to use hotkeys to quickly switch between two common configurations. First of all, with your monitor unplugged from a second screen, open up a terminal and type:

```
dconf watch /com/ubuntu/user-interface/scale-factor
```

2 Tweak your settings

Leave that terminal open and go to System Settings > Displays. You should see a line that says something like {'eDP1': 16}. The number 16 means that the scale factor is 2:1 for the built-in display. An 8 means 1:1 etc. Take a note of what it says, and plug in your second screen. Now change the scaling factor and resolution to something comfortable for both screens (you'll probably have to reduce the resolution of your built-in screen).

3 Create hotkeys

You'll see a new reading in the terminal that will look like {'DP1': 8, 'eDP1': 8}. Make a note of this. Now go to System Settings > Keyboard and open up the Shortcuts tab. Create two custom shortcuts; set one key combination for the higher scaling factor, e.g. `dconf write /com/ubuntu/user-interface/scale-factor {'eDP1': 16}` and a second for the lower one. Now you can quickly change scale factors when you want.

Gaming: Set up your gamepad

Gaming on Linux has truly come of age and **Mayank Sharma** shows you how to make the most of gaming sessions by adding a gamepad.



Our expert

Mayank Sharma

is a technical author and former contributing editor at Linux.com but now spends his time playing Linux games all day in his underpants.

The vast majority of games for your PC will have been written with the mouse and keyboard in mind, but sometimes you can't beat playing a game using a proper gamepad, whether you prefer an old-school joystick or console-like controllers, and in the case of some games, such as *Hyper Light Drifter*, they are designed for gamepads.

Ubuntu is rapidly becoming a popular choice for gamers thanks to its official *Steam for Linux* support and its lean demands on system resources—and when it comes to using a gamepad with Ubuntu, the good news is that the vast majority of PC game controllers should work out of the box while many USB console controllers (particularly Xbox) should also be recognised. Support for these is baked into Ubuntu in the form of the XPAD kernel driver. It offers decent basic support for key functionality and installation involves nothing more than plugging in your game controller.

Test and configure

Of course, plug and play is all well and good, but how do you verify the controller has been detected and is working correctly? The answer lies with the *jtest-gtk* tool. Install this via the *Software Centre* by searching for 'joystick' (it's the first entry in the list). Once done, it should appear as a shortcut on the launcher—double-click it to start.

You should see your game controller's exact model listed. If a generic controller is listed instead alongside your keyboard, try rebooting Ubuntu with the controller plugged in. When your controller is correctly identified, click 'Properties' to test it. A new window will open split into two sections: Axes for tracking joystick movement, and Buttons for – wait for it – registering button presses.

As you move or click buttons, you should see the screen update accordingly—if things don't work quite as you expect, click 'Calibration' followed by 'Start Calibration' to run through a wizard that should help you get your controller set up correctly. You'll also see a 'Mapping' button on the main screen, which will enable you to change which buttons do what. Some knowledge of how gamepad buttons work is required here to correctly interpret which button does what, but don't worry—there are more user-friendly and intuitive mapping tools out there.

As things stand, your gamepad should work with any game that supports controllers. All you need to do is find one—for starters, open the *Ubuntu Software Centre* and search for 'SuperTuxKart'. This racing game should work out of the box with your controller. If you're a Steam user, you need to set up your gamepad prior to playing any games with

it. Launch *Steam* and click the game controller icon to switch to Big Picture mode. Click 'Settings' followed by 'Add/Test under Controller'. Your controller should be detected, select it and you may be prompted to map your controller using the wizard if it's not already known to *Steam*. Do so and if you make a mistake, simply use the mouse to select any previously set control to change it. Once complete, you'll be invited to upload the controller for others to use – accept this.

Once these steps have been performed, you should find your controller works perfectly with any Steam game—before playing, however, check the game's Options screen to see if gamepad support has been enabled.

Using AntiMicro profiles

You'll find that many Linux games don't directly support controllers – instead, they assume you'll be playing with a mouse or the keyboard (or combination of both). This isn't a deal-breaker, thanks to the fact you can emulate key presses and mouse movements on your gamepad, allowing you to use it with just about any game.

One of the best ways to do this is *AntiMicro Portable* (<https://sourceforge.net/projects/antimicroportable>), which as the annotation (*right*) reveals makes the emulation process as straightforward as possible. You'll need to install it through the Terminal with the following commands:

```
sudo add-apt-repository ppa:milarimogard/webupd8
sudo apt-get update
sudo apt-get install antimicro
```

Once installed, open the Dash and type 'antimicro' to locate and launch it. First, you need to click 'Controller Mapping' to map the expected buttons and joystick controls to your controller. You'll see an image of a controller appear, with each button highlighted in turn—which is far easier to follow than *jtest-gtk*'s somewhat cryptic approach. Press the button and work through the list. Again, if you make a mistake, just click on the incorrect entry and press the correct button to fix it.

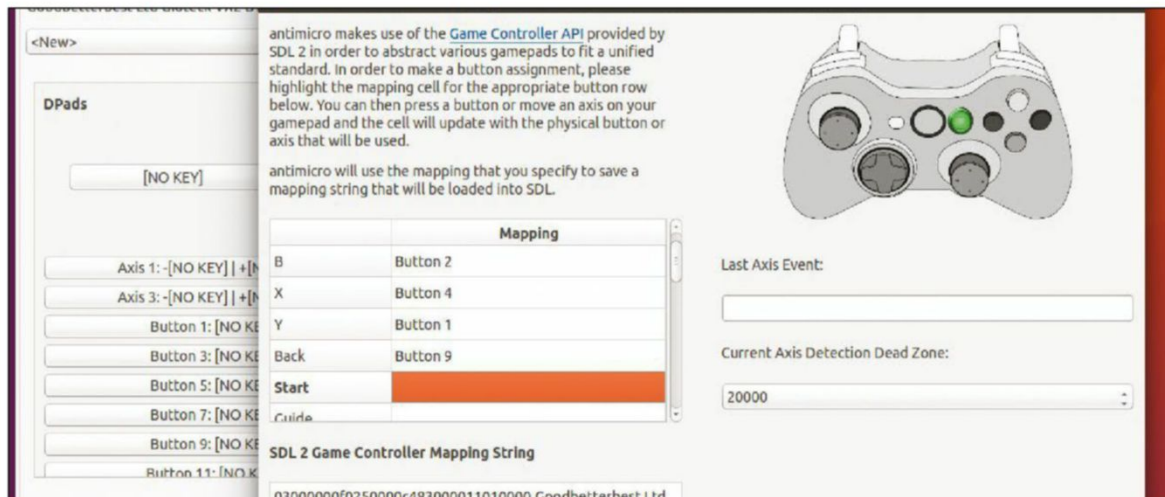
Once done, *AntiMicro* stores key and mouse settings in profiles and sets. You could, for example, store multiple game settings in a single profile by allocating each to a separate set or you can set up multiple sets for a single game and be able to quickly change the way your gamepad controls the action within a game, e.g. for different classes in a team game.

You'll obviously need to know what controls your target game uses before you can attempt to map them to your controller. Typically launching the game alongside *AntiMicro* and locating the instructions or settings will give you the

Quick tip



If you have an Xbox controller, visit <http://pingus.seul.org/~Egrumbel/xboxdrv> for details of an alternative driver (Xboxdrv), which has support for additional features such as force feedback.



AntiMicro makes it easy to remap your controller and get the experience you're looking for from your games.



Quick tip
When allocating key assignments in *AntiMicro*, click 'Advanced' if you need to assign more complex actions to buttons, such as holding a button for a certain length of time before it's activated.

information you need. Otherwise, try the particular game's web page for a wiki.

Once set up, you can launch the game in question and you'll see that *AntiMicro* now translates your controller's movements into the equivalent key presses and mouse movements (and clicks).

Adding profiles

Going forward, make sure you're running *AntiMicro* alongside any game you wish to emulate—pick your profile and/or set, then start the game in question.

If you find the controller is too sensitive (or not sensitive enough) for you there are plenty of ways that you can tweak specific controls, particularly when assigning mouse actions to your controller.

Before setting up a profile, take a quick trip to the site at

<https://github.com/AntiMicro/antimicro-profiles/tree/master/applications> where kindly users have shared their own custom *AntiMicro* profiles for a wide number of games. These exist in XML format and need to be downloaded in the correct manner in order to work with *AntiMicro*.

Scroll down the list, and if you see the game you're currently playing – the brilliant first-person shooter *Open Arena*, for example – then click it to locate the XML file within. Click the XML link to reveal the code and click 'Raw' to view the code directly. Right-click this and choose 'Save page as' to save the file to a suitable folder on your hard drive (say an *AntiMicro* Profiles folder inside **Documents**, for example).

Once you've done this, open *AntiMicro* and click the 'Load' button. Select the XML file and you should see the window update with the correct key and mouse assignments for the game in question. **LXF**

Exploring the antimicro interface

Joystick list

Each controller is allocated its own tab. Click on a tab to select the controller you wish to configure.

Save/select profile

Select a previously loaded profile from the drop-down menu, or click 'Load' to open one. Click 'Save' to update the current profile or 'Save As' when setting up a new one.

Mapping controls

Click on a specific control or button to assign it a key or mouse movement (or click) from the pop-up menu.

Controller mapping

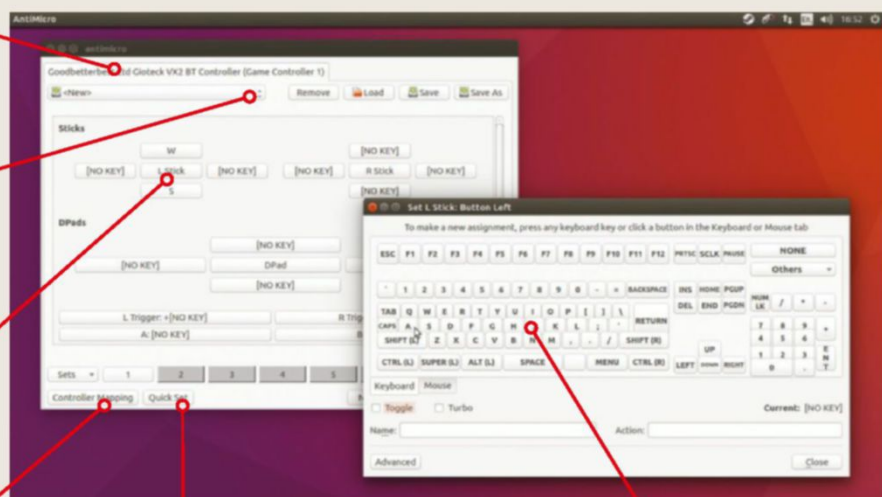
Before using *AntiMicro* for the first time, click this option to map *AntiMicro* to your controller's specific joystick axes and buttons.

Quick Set

Click this, move your controller's joystick or press a button and choose what key or mouse action that you want to assign it to.

Select your key

When you click a control in the main window, this allows you to select the key it's mirroring. Switch to the Mouse tab for mouse-related commands.



» **We're #1 for Linux!** Subscribe and save at <http://bit.ly/LinuxFormat>

» **Dual boot** Learn to dual-boot both Windows and Linux on the same PC

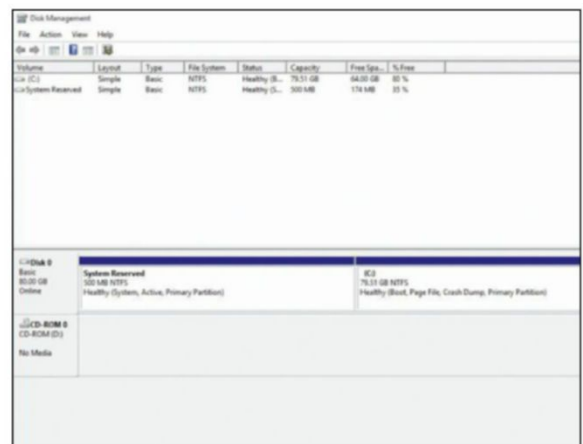
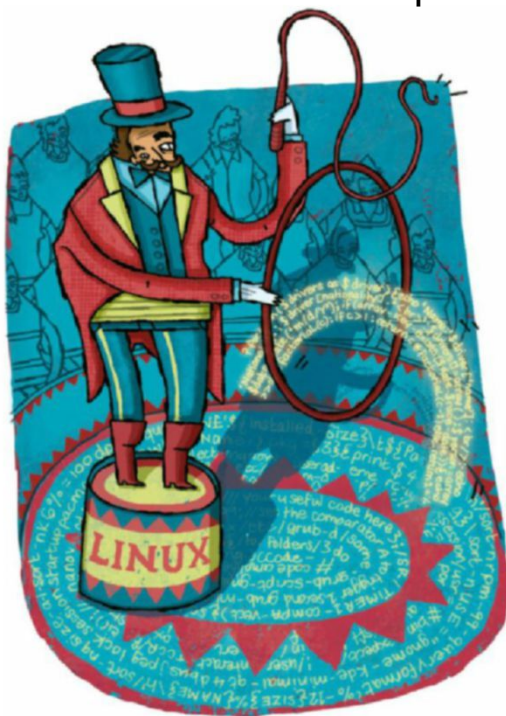
Dual boot: Set up two OSes

Afnan Rehman dives deep into the world of dual-boot systems to bring you the simplest solution to the two OS problem.



Our expert

Afnan Rehman is a student, Linux tinkerer and general computer geek who breaks everything first so you don't have to.



» The Disk Management tool provides an easy to understand graphical interface for partitioning and managing storage on all disks that the OS has access to.

computer's hardware resources. The fact is, virtual machines will severely restrict the amount of resources that you can use. When you implement a dual-boot system, you're able to boot as if Linux is the main OS and Windows weren't there and vice versa. That gives you access to all the processing cores, all system RAM and all expansion cards that the computer has to offer, making it easy to run intensive programs without slowdown. You also get access to all disks and externally connected devices such as USB devices at full speeds (as long as the OS supports them, of course).

How dual-booting works

For the sake of brevity, we will assume you are able to install Windows without a detailed explanation. It's safe to say that if you've ever installed a Linux distro, you are more than ready to install Windows on your own. If you're a newcomer to Linux – welcome! – and you're likely to have Windows installed on your current system (given that the Windows family still has around 90% of the market) and can proceed directly to creating a dual-boot system as a fresh reinstall of Windows isn't required. If you haven't tried installing Windows before, the process is simple and there are many guides online to help you. The installation media is usually sold on a disk or USB drive and can be often purchased online or in store. Once you insert the media and boot the computer, a series of steps will guide you through the installation.

The version of Windows that you choose to install shouldn't matter, whether it's Windows 7, 8 or 10. The type of

We live in a world where multiple operating systems hold a share of the desktop market and applications aren't always available on every platform that you want them to be, it's often necessary to have more than one OS available ready to boot. Virtual machines offer only a partial solution to this problem as do compatibility layer technologies such as *Wine*. The solution is to dual-boot two OSes on the same machine.

The boot managers of modern day OSes can manage multiple systems and allow the user to choose which one to boot at startup. This way you can have Windows OS and a Linux distro, so that you can keep the beautiful experience that is Linux while also accessing applications that still only work well on for work or play. (Not every game supports SteamPlay on Steam for Linux, although we're getting closer). In this tutorial, we'll focus specifically on dual-booting a Linux distribution (distro) with Windows. We will be using Ubuntu. However, the steps in this tutorial can be adapted to other Linux distros.

First, let's talk a little bit about why we dual boot. There are many reasons people pursue a dual-boot solution, the key reasons being performance and compatibility. As most users will tell you, it's best to run things natively with access to the

Quick tip

Make sure to disable Fast Startup before installing Ubuntu to avoid issues with GRUB detecting the Windows installation. You can disable it in the Windows power management advanced settings under 'Shutdown settings' where you can uncheck 'Turn on fast startup'.

What to do when you mess up

Should you need to repair the Ubuntu bootloader after trying to install a dual-boot system, the Ubuntu wiki has some great instructions for helping to repair the bootloader. The link is <https://help.ubuntu.com/community/RecoveringUbuntuAfterInstallingWindows>. The recommended way would be to use an Ubuntu live CD or USB that you created when first installing Ubuntu. The simple graphical way is to insert your disk or flash drive, reboot the

computer and boot into the live CD environment. From this environment, you can choose the *Boot Repair* tool which is built into the Live CD.

The boot repair guides you through solving more frequent problems, such as those associated with dual-boot and can also generate error reports that you can share on forums or email to your genius friends who are more Linux savvy to get help with the issue. For recovering Ubuntu, you can click 'Recommended Repair'

and allow it to work its magic. Then all you have to do is reboot and the usual boot menu should appear. If not, you can hold left Shift key while booting to select between the two operating systems. You can also try to repair it by opening a terminal. When you are given a standard *Bash* prompt, type:

```
# sudo grub-install /dev/XXX
```

Replace **XXX** with your Ubuntu install device (eg. **/dev/sdb**).

program that's used to get the system started after turning it on, either legacy BIOS or EFI boot, may matter in some cases but on most modern systems this won't cause any major issues. Let's get some basic terminology out of the way.

EFI stands for Extensible Firmware Interface, and most modern system partitions are in this format, which adheres to the UEFI (Unified Extensible Firmware Interface) standard. UEFI replaces the Basic Input/Output System (BIOS) that most PC devices used before. This is important mostly for more advanced users who wish to manually set partitions for more customised setups. Since most modern systems use UEFI, we'll work on the assumption that you will too.

Each hardware manufacturer has a different implementation of UEFI and each has its own unique quirks. Most have the same basic functions and structure, which allow for the installation of different operating systems. However, Different UEFIs can manifest major problems for some and instant success for others. Different button settings, boot menus, and startup sequences can cause problems when you are trying to mess with the boot sequence. It's best to research your manufacturer's UEFI manual beforehand and familiarise yourself with the settings before you try to change anything. Most system boot sequences will feature a splash screen where you can press a function key to gain access to boot settings menus (for example, it's F9 for HP, F12 for Dell and Lenovo, F8 in Amibios and F11 in an Award BIOS) and you'll need to know which key it is to open which menus and which settings to change. Knowing these things beforehand can save you a lot of headache should the need for troubleshooting arise.

Partition standards

You'll also hear the acronyms GPT and MBR thrown around. These terms relate to how partitions in the hard drive are organised and recorded. MBR, which stands for Master Boot Record, refers to the information stored in the first sector or partition of the hard drive that identifies how and where an operating system is located so that it can be loaded into the computer's main storage or RAM.

GPT, which stands for GUID Partition Table, is similar in function to MBR in that it stores partition information for the hard drive. This information includes where the partition starts and ends so that the OS knows which sectors belong to which partition and which ones are bootable. MBR, which was first introduced back in 1983, can only handle drives that are up to 2 terabytes in size and can support up to four partitions per

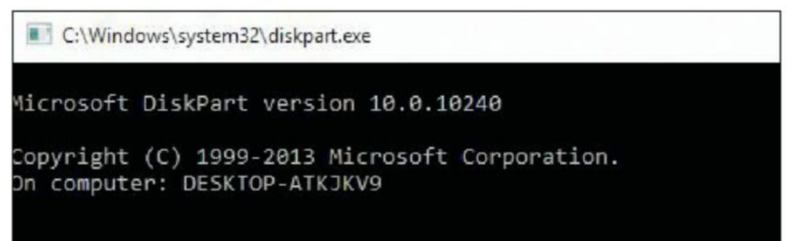
drive. GPT is a newer standard that has been gradually replacing MBR and handles drives that are far larger and allows for a nearly unlimited amount of partitions. If you have Linux the *GRUB* bootloader, which handles OS booting just like the Windows boot loader, is often stored in the MBR or GPT. It's important to note that Windows can only boot a GPT drive on a UEFI-based machine. All current versions of Windows, Linux and Mac OSX support GPT and it's generally recommended to use this scheme on any new installs for the sake of compatibility and to avoid issues that MBR might bring.

There are benefits and drawbacks to either installing Windows or Linux first and every system is different. In this guide, we will be taking you through the process having Windows installed first. It's possible to install Ubuntu first, but there are far fewer issues when having Windows installed first, as the Linux *GRUB* bootloader can adapt easily to accommodate Windows. In contrast, the Windows bootloader will almost always disregard the presence of another existing OS, often overwriting the existing bootloader and rendering the other OS useless until it's repaired.

It's important to note that while most of Linux distros are good at handling dual-boot situations, there's always room for something to go wrong. We highly recommend performing a backup of any existing data to a separate storage device just in case the worst happens. Any files, settings or applications can be backed up in a variety of ways that will allow you to pick and choose what to save for later. It would be best to save this backup on a separate physical device, such as a different hard drive, a different computer or possibly cloud storage if you have enough space and a fast enough connection. The simplest way to back up everything you need to back up is to make a recovery disc which will back up Windows itself and set up a File History backup to take care of »

Quick tip

Always keep a copy of the Ubuntu live CD and Windows installation disc on hand. These two discs are essential and will save your bacon should the worst happen. The tools and functionality will help reverse most common issues and can be used to help recover data in a pinch.



» The *diskpart* tool allows you to accomplish partitioning and storage tasks with a command line interface for more advanced users and those who might wish for more custom settings.

» **Want even more Linux?** Grab a bookazine at <http://bit.ly/LXFspecial>

personal files. There are wizards located in the Control Panel that will guide you through both of these backup processes. The recovery disc will help you restore Windows system and boot files in case something goes wrong with the operating system. File History will automatically back up personal files to a designated device in case the worst comes to pass and your entire drive partition gets corrupted.

Now that you understand the basics, let's get to work. The first thing you'll need to do is choose your distro of Linux that you want to install. For this tutorial, we're using Ubuntu 16.04. The most important thing you will need to start is a copy of Ubuntu live CD on a USB flash drive or compact disc. You can download your preferred version of Ubuntu from its website at www.ubuntu.com/download. Here you can download an ISO file that you can use to create a DVD or bootable USB flash drive. We recommend creating a bootable flash drive as it's more versatile than a DVD, especially with optical drives getting phased out of many newer systems to save space, weight and power. Once you have a flash drive made you can move on to getting your partitions in order to prepare for the installation.

Of course, now you will only be able to resize the Windows partition if there's a decent amount of available space on it, so make sure you have room before you commit to undertaking this project. If your computer is running multiple hard drives you also have the option of installing Linux on an entirely different drive, leaving your Windows partition untouched. Now before you run out and purchase a brand new hard drive, I'll remind you that the two installations are perfectly capable of running in harmony on different partitions of the same physical drive.

Partitioning methods

There are a few ways to handle partitioning, so I'll explain the one that assumes you have Windows already installed. If you already have Windows installed, the easiest method is to just use the *diskpart* utility built into Windows to help with this process. Every drive is different, and it's likely you'll have multiple partitions already existing on the hard drive, such as recovery, OEM and primary etc. Be careful resizing or deleting

these partitions as messing with the wrong one will lead to major issues. Usually, the partition labelled 'Primary' is the largest and is where most of your files and data are held. Any partitions labelled 'System' or 'OEM' should not be modified at all, as those hold Windows system files and cannot be changed, lest you risk the entire system being corrupted. The Primary partition could be resized to give you space for your Linux installation. The easy way to do this is to go into the Disk Management application, find your drive with the primary Windows installation, right-click the partition and select 'Shrink Volume'. After that, just follow the onscreen instructions and you're good to go. Now you'll see a section of the drive with a black bar above it labelled as 'Unallocated'.

This is where your shiny new Linux installation will go. Alternatively, if you're feeling brave you can lose any partition named 'Recovery' and install there. However, make sure you know what you're getting yourself into when you do this.

Now, you can restart and boot from a USB or disk into the Ubuntu live CD by holding down the Shift key while rebooting. On booting, hit the appropriate function key to load up the boot menu, where you can select which device to boot from. Find your flash drive and hit the Enter key to commence the boot process. You'll be loaded into the Ubuntu live CD environment with its basic feature set. Here, you can start the installation process.

Now follow the onscreen instructions to install Ubuntu. Make sure to choose options that would allow for the Ubuntu install to exist alongside the existing Windows installation. Don't try to overwrite it, because that would be disastrous. This step will involve you as the user deciding how much space to allocate for your Linux installation. You've already set aside space for the Ubuntu installation in Windows Disk Management, but you also have the option to use more or less space depending on how you want to do things. In Ubuntu, there are two ways to install alongside Windows. First, there's an option labelled 'Install Ubuntu alongside Windows' which will handle everything for you. Some other Linux distros, such as those using *Calamares* installer, offer similar options to simplify the process. Otherwise, you can also make and resize partitions using the installer. If you choose this route, you can select 'Something Else' when prompted for installation type. You can resize the partitions based on your needs and use case. If you're mainly using Linux to experiment, you may not need as much space as if you were, for example, heavily invested in Linux gaming or dealing with other large amounts of data. Make sure that the boot loader remains the Windows Boot Manager.

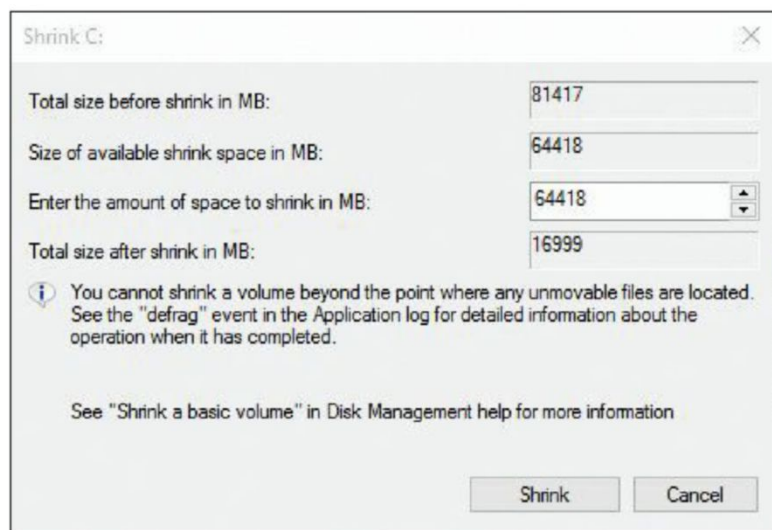
The installation process itself will vary depending on the Linux distro that you want to use. For Ubuntu, the process is a fairly simple step-by-step walkthrough. You can choose to install updates during the installation, however, you can skip this if you have a poor internet connection or just want to install them later. At the very end, you'll be asked to restart.

At this point, you're done and can start using your Ubuntu Install. Switching between OSes would involve a simple reboot. When booting you'll be given the option to load either Windows or Ubuntu, usually with a default option that will automatically boot after several seconds. If you're not receiving these options, try some basic troubleshooting by retracing your steps.

Sometimes you may find that you need to remove the Linux install for whatever reason. When removing Linux from

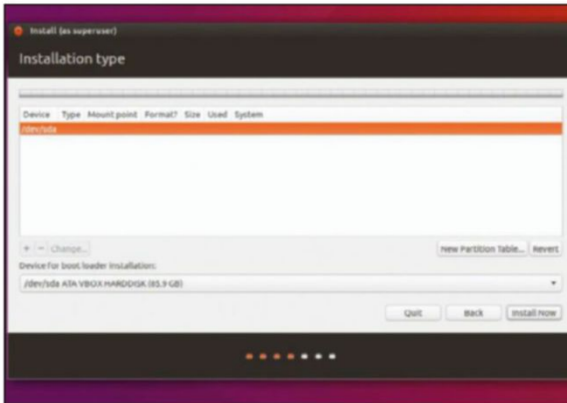
Quick tip

When backing up your data, make sure your backup storage is separate, reliable, and spacious. By keeping it physically separate from the main device, and making sure it is reliable and spacious enough to hold all of your data, you will reduce potential issues with the backup failing.



➤ The shrink volume dialog box in Disk Management allows you to shrink a partition size, and also warns you if your desired shrinkage is too small.

➤ **Get print and digital subs** See www.myfavouritemagazines.co.uk/linsubs



► **The manual partitioning tool in the Ubuntu installer is similar to many partitioning GUI tools.**

the dual-boot process you must be very careful so as not to break your bootloader and, in turn, create a massive headache and possibly render the Windows install unusable.

Removing Linux

To start, you'll need your Windows installation or recovery CD or an Ubuntu live DVD. To remove the *GRUB* bootloader, open the command prompt from the installation or recovery media environment. Next, type:

```
> bootrec /fixmbr
```

Reboot and boot into Windows. Then move onto using the Ubuntu live CD. Boot from the CD and choose *Boot-Repair*.

If it isn't installed then you can try to install through a few simple terminal commands that you can type one by one:

```
$ sudo add-apt-repository ppa:yannubuntu/boot-repair
```

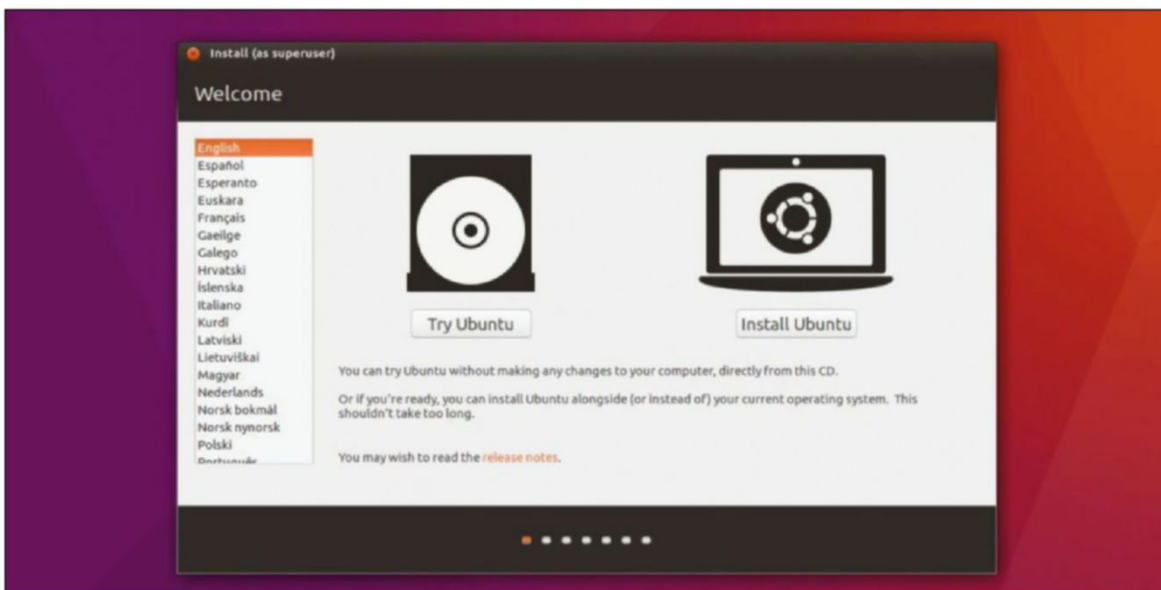
```
$ sudo apt-get update
```

```
$ sudo apt-get install boot-repair
```

Next, simply **boot-repair** and click the 'Recommended Repair' option.

Last, to remove the Ubuntu partitions you'll need to head back into Windows, reboot and boot into Windows and open Disk Management by searching for it in the Windows menu. In disk management, right click the Ubuntu partitions and select 'Delete'. This is the point of no return so make sure you're good to go before you click that button. If you plan to just keep using Windows then right click the Windows partition and select 'Extend Volume'. This will increase the size of the Windows partition to fill up the free space. That should be all that is needed to restore the machine to using only one operating system.

Now that you know how to add and remove operating systems to and from a computer in a dual-boot situation, you are ready to tackle your own project with other distros and combinations. As a final note, there's also the possibility of adding more OSes to your boot sequence, creating triple-boot or multi-boot systems. Adding systems essentially follows the same basic steps of partitioning and adding as space allows to the hard drive. With more operating systems comes more versatility and using the bootloader to your advantage can allow you to truly make your computer your own and design an environment suited to your needs. **LXF**



► **The Ubuntu live environment allows you to try Ubuntu right off of the disc, so you can use it to test some features and prepare your machine before writing anything to disk.**

Re-ordering the Boot Manager

You'll notice that when you boot from a shutdown state there's a certain order to things. The Windows operating system or the Linux one will be given priority and unless you press keys during startup you'll end up automatically booting into one system or the other, whether you wanted to or not.

This is a common issue that's easily remedied. You can use the EFI Boot Manager tool that you can access from an Ubuntu Live CD. Once in the Live CD environment open a terminal and install

EFI Boot Manager with the following:

```
$ sudo apt-get install efibootmgr
```

Next use the following command to find out the current boot order situation:

```
$ sudo efibootmgr .
```

You'll get an output that looks something like this, with the boot order listed:

```
BootOrder: 0004,0001,0002,0005
```

```
Boot0001 Windows Boot Manager
```

```
Boot0002 Network
```

```
Boot0004 ubuntu
```

```
Boot0005 Hard Drive
```

```
Boot0006 CD/DVD/CD-RW Drive
```

To change the boot order, we will have to use the numerical codes which correspond to the boot device that we want first. For example, in this case we want the system to try to boot from CD first, followed by Ubuntu and Windows. So, we will use the following command to change the boot sequence:

```
$ sudo efibootmgr -o 0006, 0004, 0001
```

That's it, you're now ready to go.

CentOS: Server made simple

Mayank Sharma shows off the community-fostered project that offers an enterprise-class distro without the enterprise-class pricing.



Our expert

Mayank Sharma

is so sceptical that he has to verify his authenticity every time he wakes up in the morning. He's also a veteran tech writer... or so he says.

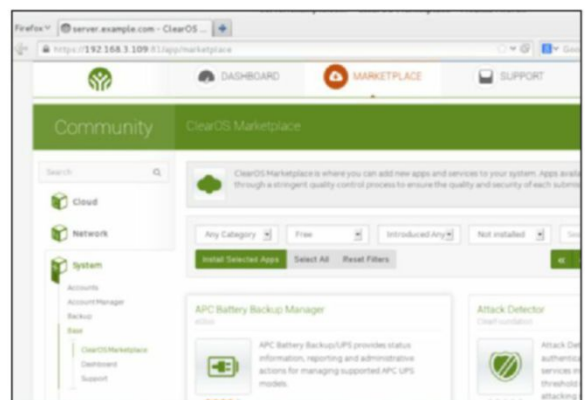
The goal of the CentOS (short for Community ENTERprise Operating System) distribution (distro) is to deliver an enterprise-grade operating system without the costs usually associated with such endeavours.

It delivers on its enterprise promise since it's compiled from open source SRPM (source Red Hat Package Manager) files from the Red Hat Enterprise Linux (RHEL) distro. RHEL is based entirely on open source software and is required by the GPL licence to release all of the source code to anyone who has a subscription. Red Hat does one better and makes its source code available to anyone.

Note however that while the source code is free, Red Hat still has a trademark over the product. That's where the CentOS developers enter the picture. The project takes the freely available RHEL source code (not the binaries), removes trademark and branding information that's owned by Red Hat and then rebuilds the packages as CentOS. So in essence you get a Linux distro that includes most of the same open source software projects that are in RHEL but can still be freely distributed without paying subscription fees.

Not just for freeloaders

While it might sound like a freeloader's paradise, CentOS is used around the world by people who need a reliable platform to deploy their apps and services. The project backs up the software with 10 years of support, which makes CentOS particularly attractive for any kind of server rollouts. However, over the years the distro has particularly become popular with hosting companies along with



» The ClearOS box marketplace includes several free (and some paid) applications and services to give wings to the barebones install. (see *Point and Click Servers* box, right.)

businesses that have in-house Linux expertise and don't want to pay for RHEL support.

The CentOS team tracks the development of RHEL and its releases are influenced by the release schedule of the upstream distro. New CentOS releases tend to trail a month behind the RHEL release date because the CentOS project has to do all of the rebuild and testing work. The distro releases security updates throughout the life of the release as and when they're available. CentOS has received some flak in the past for release delays. But the project's 2014 partnership with Red Hat, which now has some key CentOS developers on its payroll, has negated that factor as well.

Kickstart a CentOS installation

You can automate the installation of CentOS (and other RPM-based distros such as RHEL and Fedora) by using what are known as kickstart files. These are text files that contain instructions for the *Anaconda* installer. The instructions vary and can include language and localisation settings, the layout of the partitions as well as authentication information for the root user. You can also use kickstart files to select package groups and individual packages that you want to install.

You can use different kickstart files for installing different types of systems, such as a web server, a mail server or a graphical desktop. What makes

the kickstart file so powerful is that it enables you to embed scripts that are executed at key stages of the install process. This means that you can automate a lot of the work that you'd normally do by hand and have the installer run all of those steps for you. For example, you can automatically restore files from a backup and modify *yum*'s configuration files to download updates from a local mirror instead of the CentOS servers.

When you install a CentOS machine, the *Anaconda* installer will save a kickstart file for that particular installation under `/root/anaconda-ks.cfg`. You can use this file to install

another system, identical to the one you've just installed. Furthermore, you can also use this kickstart file to customise and create your own files. The most convenient way is to use the graphical *Kickstart Configurator* tool (see the *last page*) that you can download and install with `yum install system-config-kickstart`. To use the kickstart file to start an installation, refer to the chapter on Kickstart installations in the RHEL documentation at https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Installation_Guide/s1-kickstart2-startinginstall.html.

Point and click servers

If you need the robustness of CentOS in a server distro that's easy to rollout and administer, grab yourself a copy of ClearOS Community Edition. One of the biggest advantages of ClearOS over other similar offerings is its larger repos of supported server software. The distro supports over 80 free services for various roles including a network server and a cloud server and more. In addition to common servers such as a directory server, database server, mail server, web server, FTP server, content filter

and more, you can use the installation as a seedbox and a Plex Media Server.

ClearOS also includes several system and network management tools for creating backups, managing bandwidth and RAIDs etc. New admins who aren't sure of the components to install can use the Feature Wizard, which helps pick services. The number of options available to you depend on whether you plan to deploy ClearOS inside a private network or as a gateway server or in a publicly accessible network.

Since it's based on CentOS, the distro uses the same *Anaconda* installer. Once the installation is complete, the distro will take you through a basic setup wizard where you'll be asked to select whether your ClearOS installation will be used inside a protected network (like an office), in a publicly accessible network (like a hotspot or a data centre) or as a Gateway server. You'll also be asked to create an account at www.clearos.com and register your installation before you can access its server apps and services via its marketplace.

Also, while the distro is 100% binary compatible with RHEL and should work on all hardware that's certified by Red Hat, as of CentOS v7 the project only puts out releases for the x86-64 architecture. The binary compatibility means that from the installation right up to the desktop, CentOS mimics RHEL in every aspect. The distro uses the Anaconda installer (see *over the page, Install CentOS*) and can be used with Kickstart to run installs across multiple machines (see *final page, Create a Kickstart File box*).

Redline the RPMs

An important aspect of administering a CentOS server is to understand its package management system and its various online repositories (repos). Together they ensure you are always running a secure and updated server.

CentOS uses the *Yellowdog Updater, Modified* (*yum*) package manager to install and update software packages in RPM format (as opposed to DEBs) from online software repositories. Furthermore you can use *yum* to check for available updates and fetch information about available packages. The `/etc/yum.conf` file comes preconfigured with options that affect how you download and use RPM packages. Here's a snippet from the file:

```
cachedir=/var/cache/yum/$basearch/$releasever
```

```
keepcache=0
debuglevel=2
logfile=/var/log/yum.log
```

The `cachedir` variable points to the location where the RPM packages are downloaded.

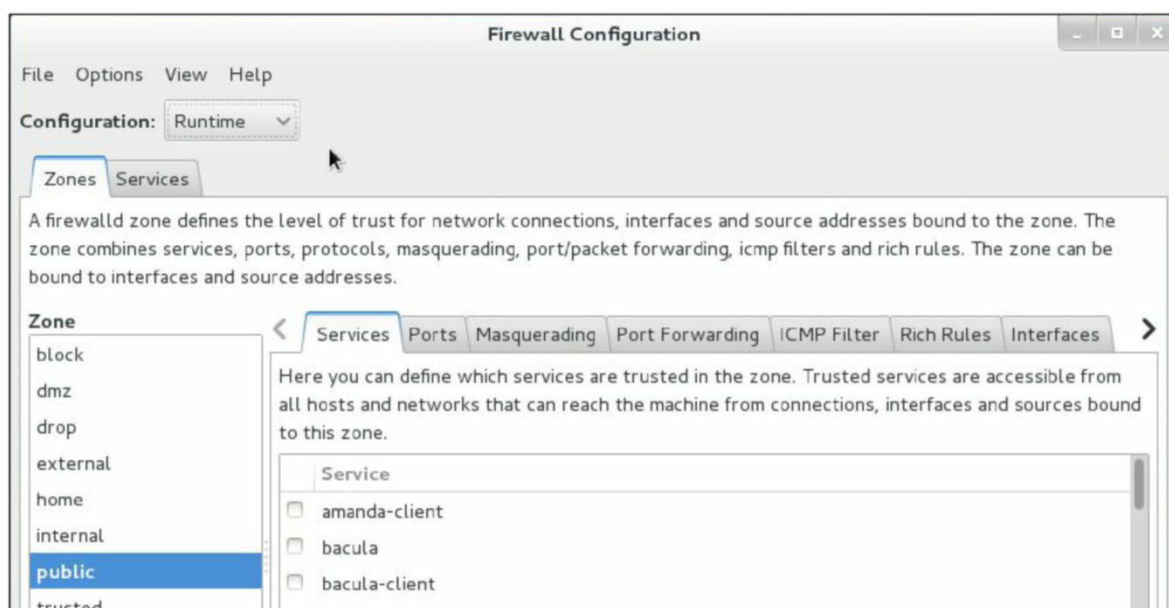
The `keepcache=0` option instructs *yum* to delete the packages after they've been installed. If you change the value of the `keepcache` variable to `1`, CentOS will keep the packages even after installation. The `debuglevel` variable can take values from 0 to 10. The default level (`2`) produces just enough information to indicate whether an operation was a success or a failure. These debug messages are logged to the specified log file, in this case `/var/log/yum.log`.

The CentOS repos

When you invoke the *yum* command to install a software package, it checks the list of configured repositories under the `/etc/yum.conf` file and in files under the `/etc/yum.repos.d` directory. Although you can add information about repos in *yum*'s main configuration file, a good practice is to list them under `/etc/yum.repos.d` in separate files with .repo as an extension, such as `CentOS-Base.repo`. This helps in managing the repos especially if you are pulling in software from lots of different sources.

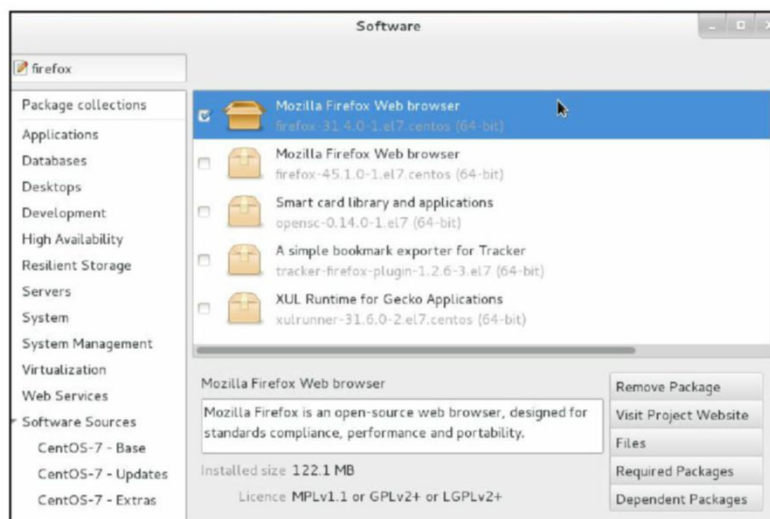
Quick tip

If you have multiple CentOS machines on the network, you can easily configure one as a local update server. This will enable you to download updates from the internet on to the update server where they can be picked up by other CentOS machines on your network.



» CentOS now uses the **Firewalld** interface instead of **iptables**. This new interface uses the concept of **zones**, each of which house services according to the level of trust.

» **Improve your Linux skills** Subscribe now at <http://bit.ly/LinuxFormat>



CentOS has several official repos. Using the default repos ensures that your CentOS installation is binary-compatible with RHEL. You can find a list of all the official repos (some are enabled and some aren't) under the `/etc/yum.repos.d/CentOS-Base.repo` file. To enable a repo, edit the `CentOS-Base.repo` file and scroll to the repository you want to enable. Toggle the repo by changing `enabled=0` to `enabled=1`.

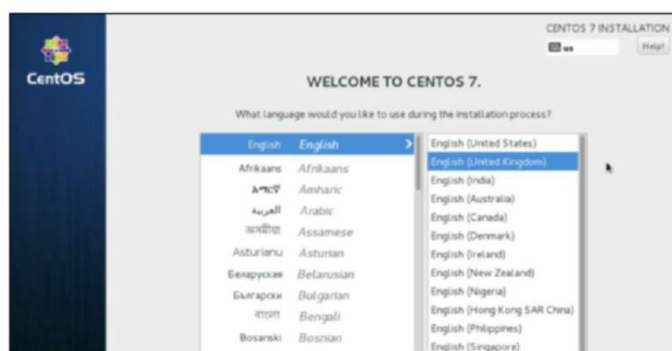
Third-party repos

CentOS is also popularly used as a business desktop distro. If you are using CentOS on the desktop, chances are you'll need a package that's not in one of the official CentOS repos, such as the Flash plugin or Google's *Chrome* web browser. In that case you'll need to enable a third-party repos.

You can use lots of third-party repos (you can see a list here: <https://wiki.centos.org/AdditionalResources/Repositories>) to flesh out your installation with all kinds of applications. However, you should know that these repos contain packages that aren't approved by the CentOS project. It's also advisable to only add the repos that you need since

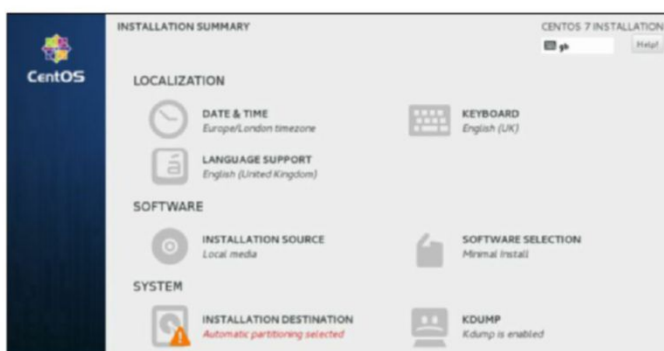
➤ Using third-party repos, you can easily flesh out a CentOS installation as a full-fledged desktop, but be aware that they aren't officially approved.

Install CentOS



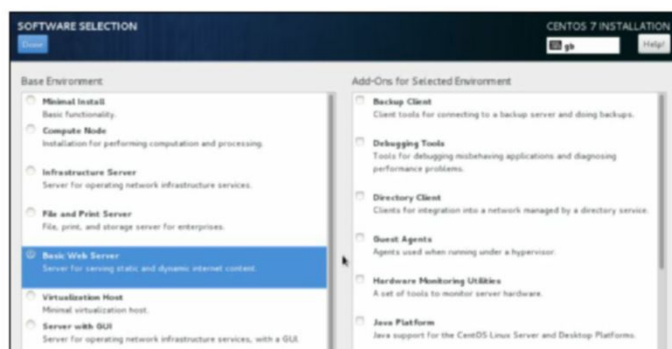
1 Select language

CentOS is available in several flavours. Besides the usual install-only mediums, the project also occasionally produces installable live discs based around the KDE and the GNOME desktop environments. All the releases use the Anaconda installer and begin the installation procedure by asking you to select the language for the process.



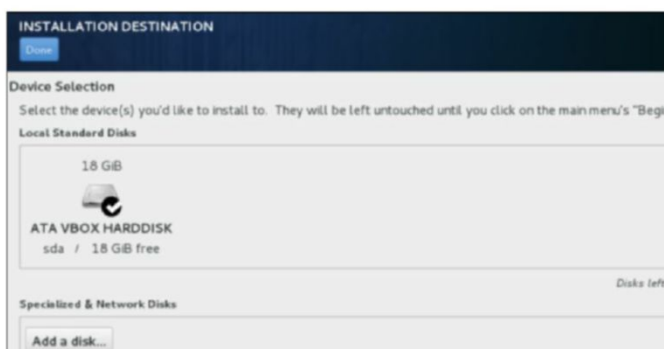
2 Installation summary

The *Anaconda* installer uses a hub-and-spoke model instead of a linear process. You're dropped at the installation summary screen from where you can configure the different aspects of the installation by visiting their respective sections. You'll return to this screen after configuring each section.



3 Select software

Click 'Software selection' to customise the list of packages that will be installed. By default, CentOS will only install basic functionality. The Software Selection window lists various types of environments on the left pane, while any available add-on for a particular environment are list on the right.



4 Select disk

After selecting the software, head to the Installation destination section and choose a disk and partitioning scheme for this CentOS installation. You can now ask the installer to copy files to the disk. While it does, you'll be asked to set a password for the root user and optionally create a non-root user as well.

» **We're #1 for Linux!** Subscribe and save at <http://bit.ly/LinuxFormat>

adding unnecessary repos can slow down the performance of `yum` and may introduce inconsistency in your system.

The Extra Packages for Enterprise Linux (EPEL) is the most recommended CentOS repo and it contains Fedora packages that have been rebuilt for RHEL. The package to add EPEL to CentOS is available in the CentOS Extras repos. Since this repo is enabled by default, you can install EPEL with `yum install epel-release`.

The power of yum

`Yum` is a very flexible and powerful package manager. If you plan to administer a CentOS installation, make sure you spend some time and familiarise yourself with `yum`. We've already seen how to use `yum` to fetch and install a package from the repos. If you have the package on your disk, `yum --nogpgcheck localinstall package-name` will install the package and automatically check and install dependencies from the repos.

Use `yum list package-name` to search the repos for a particular package. If you don't know the name of the package, you can search for a string in the name, description, and summary of all the packages with `yum search keyword`. You can also use `yum provide filename` to search packages

that provide a package or a library. If you have configured third-party repos, you can use `yum list extras` to see a list of packages that were installed from repos outside of the main CentOS repository.

Similarly, updating a CentOS installation is also fairly straightforward with `yum`. Use `yum check-update` to check for available updates. While a simple `yum update` will install all available updates, you can update a particular package with `yum update package-name`.

Run `yum clean packages` regularly to ensure the packages are cleared out from under the `/var/cache/yum` directory. If `yum` throws a tantrum while you're installing packages, you can refresh the metadata information about the packages with `yum clean metadata` or clear the whole cache with `yum clean all`.

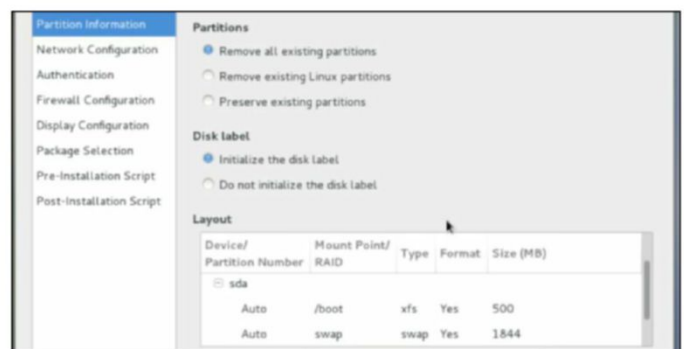
If you need some hand-holding, tap into the expansive CentOS community over forums, mailing lists and IRC. The DIYers among you will definitely appreciate the large amount of documentation hosted on the project's website and from other third-party sources and books. While the project itself doesn't have a formal paid support structure, there are a number of companies that support CentOS in a professional setting. **LXF**

Create a kickstart file with Kickstart Configurator



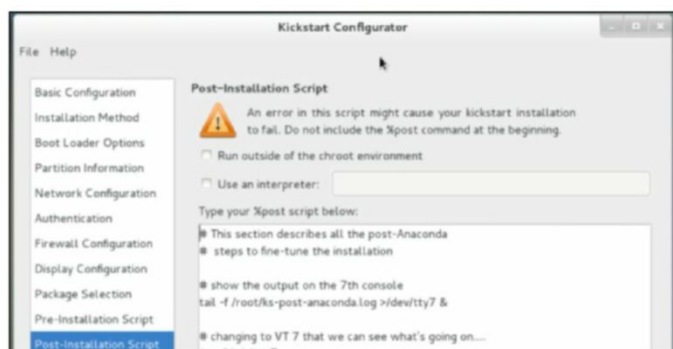
1 Basic configuration

The application has a simple intuitive layout. Browse through the sections listed on the left, each of which caters to a different aspect of the installation process. You can choose the language, specify a password for the root user, configure network devices, select an authentication mechanism, enable and configure the firewall etc.



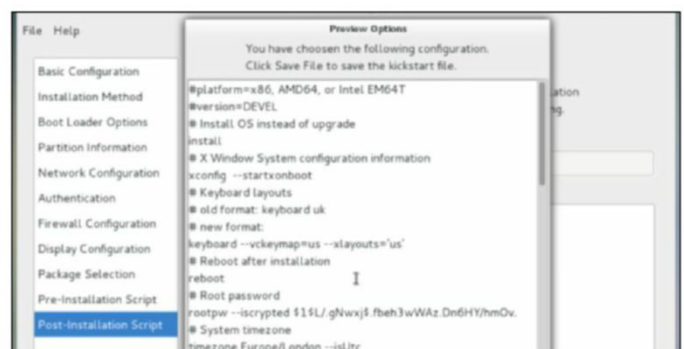
2 Disk layout

As with a manual installation, partitioning requires utmost care and attention. You can ask the kickstarter file to either preserve the existing partitions or clear the disk and specify a custom layout from under the Partition Information section. However, note that the tool doesn't allow you to create LVM partitions.



3 Running scripts

You also get textboxes where you can copy-paste scripts that are executed before the installation process begins as well as right after it ends. The commands run in the pre-installation section are provided by `busybox-anaconda`. You can also specify an interpreter to parse the script such as `/usr/bin/python3`.



4 Import and export

When you are done creating your kickstart file, or even during the process, you can review the contents of the generated kickstart file by heading to File > Preview which displays the contents in a new window. You can also import an existing file by heading to File > Open and pointing to the existing kickstart file.

psutil: A guide for sysadmins

Mihalis Tsoukalos teaches you how to use the psutil module to examine your Linux system and access information on processes and usage.



Our expert

Mihalis Tsoukalos

is a Unix administrator, a programmer, a DBA and mathematician. When not writing articles, he's knee-deep in learning new programs to (write about). You can reach him at www.mtsoukalos.eu and @mactsouk.



The main advantage of psutil, the Python module that provides information on system processes and use, is that it's cross-platform, which means that the code will work without changes on all operating systems. After you read this tutorial and see the practical examples, you'll know how to use the psutil module in your own projects, especially if you are a Linux system administrator or a developer of a complex application. In order to install psutil for Python 3 you'll need to perform one of the following two actions:

```
$ sudo apt-get install python3-psutil
$ pip3 psutil
```

The following interaction with the Python 3 shell will verify that psutil is properly installed:

```
>>> import psutil
>>> print(psutil.__version__)
5.0.0
```

This means we'll be using version 5.0.0 in this tutorial.

The basics of psutil

The most simple task you can perform with psutil is finding out the number of processes on your Linux machine:

```
#!/usr/bin/env python3
```

```
2. mtsouk@mail: ~/docs/article/working/psutil.LXF221/cod
code$ ./processID.py 10573
Checking process with PID 10573
Parent process ID: psutil.Process(pid=4021, name='master')
sleeping
puids(real=105, effective=105, saved=105)
pgids(real=108, effective=108, saved=108)
code$ ./processID.py 11772
Process does not exist: 11772
code$ ./processID.py 7179
Checking process with PID 7179
Parent process ID: psutil.Process(pid=7174, name='sshd')
sleeping
puids(real=1000, effective=1000, saved=1000)
pgids(real=1000, effective=1000, saved=1000)
[<psutil.Process(pid=7180, name='bash') at 140173242105136>]
code$ ./processID.py 7174
Checking process with PID 7174
Parent process ID: psutil.Process(pid=3353, name='sshd')
```

Here we see the kind of output you should expect from the processID.py script.

```
import psutil
nPlist = 0
for proc in psutil.process_iter():
    nPlist = nPlist + 1

print(nPlist)
print("Output format:", proc)
```

The program uses the `psutil.process_iter()` method to iterate over the processes of your Linux system and count them. If you decide to execute **simplePS.py**, you will get the following output:

```
$ ./simplePS.py
145
Output format: psutil.Process(pid=27841, name='apache2')
```

The first line from the output (above) is the number of processes whereas the second line shows that the `psutil.process_iter()` iterator returns objects that belong to the `Process()` class as defined in the psutil package. Please note that if you execute **simplePS.py** with root privileges you may get a higher number of processes. Also, remember that the Python 3 code of **simplePS.py** is portable and therefore can be executed on other Unix machines without any changes!

Uptime

This section will use the psutil module to emulate the job of the `uptime` command line utility. The important Python 3 code in **myUptime.py** is the following:

```
bootTIME = psutil.boot_time()
```

Quick tip

The psutil module can replace a host of command line utilities, such as ps, top, lsof, netstat, ifconfig, who, df, kill, free, nice, iotop, iostat, iotop, uptime, pidof, tty, taskset and pmamp!


```
2. mtsouk@mail: ~/docs/article/working/psutil.LXF221/code (r
status='NONE', pid=None), sockn(fd=-1, family=AddressFamily.AF_INE
.SOCK_DGRAM: 2>, laddr=('0.0.0.0', 41807), raddr=(), status='NONE',
family=AddressFamily.AF_INET: 2>, type=SocketType.SOCK_DGRAM: 2>,
', 123), raddr=(), status='NONE', pid=None), sockn(fd=-1, family=Ad
0>, type=SocketType.SOCK_DGRAM: 2>, laddr=(':::', 61307), raddr=(),
), sockn(fd=-1, family=AddressFamily.AF_INET6: 10>, type=SocketType
=(':::', 123), raddr=(), status='NONE', pid=None), sockn(fd=-1, famil
X: 1>, type=1, laddr='private/bounce', raddr=None, status='NONE', pi
amily=AddressFamily.AF_UNIX: 1>, type=2, laddr='/run/systemd/notify
NONE', pid=None), sockn(fd=-1, family=AddressFamily.AF_UNIX: 1>, typ
ne, status='NONE', pid=26849), sockn(fd=-1, family=AddressFamily.AF
etType.SOCK_DGRAM: 2>, laddr=('fe80::f03c:91ff:fe69:1381', 123), rad
id=None), sockn(fd=-1, family=AddressFamily.AF_INET6: 10>, type=So
, laddr=(':::', 653), raddr=(), status='NONE', pid=None), sockn(fd=-1
.AF_UNIX: 1>, type=1, laddr='private/ucp', raddr=None, status='NONE
-1, family=AddressFamily.AF_UNIX: 1>, type=2, laddr='/run/systemd/j
one, status='NONE', pid=None), sockn(fd=3, family=AddressFamily.AF
', raddr=None, status='NONE', pid=26849), sockn(fd=-1, family=Add
type=SocketType.SOCK_STREAM: 1>, laddr=('0.0.0.0', 25), raddr=(),
```

» Here's the wall-of-text output from the netPS.py Python 3 script which psutil can use to get network-related data.

```
NOW = time.time()
s = NOW - bootTIME
```

So you should use the `psutil.boot_time()` method to get the time your Linux system booted and `time.time()` to get the current time in order to find the difference between them in seconds. Then you'll have to convert that amount of time into days, hours and minutes. Executing `myUptime.py` generates the following output:

```
$. /myUptime.py
Uptime: 93 days, 3 hours, 39 minutes and 38 seconds!
```

As you'll probably appreciate, the use of the psutil module is mainly a process of using the right methods to extract the desired information.

Process ID intel

A core psutil task is how to get information about a given process when you know its process id. The good thing with psutil is that it gives you the desired information easily without having to deal with too many function calls provided that you know what you are looking for! Performing the same task using Linux command line utilities is more difficult. The Python 3 code of `processID.py` shows this:

```
#!/usr/bin/env python3

import os
import sys
import psutil
if len(sys.argv) == 2:
```

```
PID = int(sys.argv[1])
else:
    print('Usage:', sys.argv[0], 'processID')
    sys.exit(0)

if psutil.pid_exists(PID):
    print("Checking process with PID ", PID)
else:
    print("Process does not exist! ", PID)
    sys.exit(0)


p = psutil.Process(PID)
print("Parent process ID:", p.parent())
print(p.status())
print(p.uids())
print(p.gids())
print(p.children())
```

It's crucial to read the input as an integer, because the `psutil.Process()` method expects to process an integer and not a string. Additionally, it's important to make sure that the process ID you want to process actually exists, which is verified with the help of the `psutil.pid_exists()` method. Once again you can see the simplicity of psutil. However, the cost of this simplicity is having to make multiple method calls. Executing `processID.py` will generate the following kind of output:

```
$. /processID.py 7180
Checking process with PID 7180
Parent process ID: psutil.Process(pid=7179, name='sshd')
sleeping
puids(real=1000, effective=1000, saved=1000)
pgids(real=1000, effective=1000, saved=1000)
[<psutil.Process(pid=11731, name='python3') at
139829741027456>]
```

The second line displays the process ID of the parent process whereas the third line shows the status of the process. The fourth line shows the various user IDs of the process and the fifth line shows the various group IDs of the process. The last two lines are the process IDs of the children of the process! Bear in mind that the children of a process might not be alive after your script has finished its job! (See *top left for an exciting example output from the execution of the processID.py file.*)

Moving on to an equally vital are we will illustrate how to use psutil to get information about your mounted file systems. The `psutil.disk_partitions()` method returns all

Quick tip 

Interpreting the output of psutil in a correctly involves a good understanding of how Linux works. Put simply, learn the required theory before using psutil!

About Linux processes

In order to successfully use system modules such as the very useful psutil, you will need to know how Linux handles processes, ports and other system properties.

A Linux process is an execution environment that contains instructions, user-data and system-data parts and other kinds of resources that are obtained during runtime. There are three categories of processes: user, daemon and kernel processes. Unless you have special privileges, you won't be able to control processes

owned by other users including processes that deal with the Linux system and support various Linux services.

Each process really has two user IDs: the effective user ID and the real user ID. Similarly, each process has two group IDs: the effective group ID and the real group ID. You might wonder what the point is of having real user and group ids? Imagine that there's a server process that needs to watch all system files, regardless of their owner, such a process must run with root

privileges because only the root user is capable of looking at any file. However, if a request to access a given file comes in from a different user (mtsouk), the server process temporarily changes its effective user ID from root to mtsouk before trying to perform the task. If the mtsouk user cannot access the file an error message will be generated. After finishing all tasks demanded by the mtsouk user, the server process will change its effective user id back to root. Most server processes work this way.

» Improve your code skills Subscribe now at <http://bit.ly/LinuxFormat>

» mounted partitions, including information about the mount point of each entry, its device name and its mounting options.

Getting disk info

The `psutil.disk_usage()` method shows information regarding the disk usage of a given partition, which is given as an argument to it. Executing **diskInfo.py** on a Debian Linux machine with one mounted partition generates the following kind of output:

```
$ ./diskInfo.py
[sdiskpart(device='/dev/root', mountpoint='/', fstype='ext3',
opts='rw,noatime,errors=remount-ro,data=ordered')]
sdiskusage(total=24769884160, used=16321609728,
free=8196128768, percent=66.5)
```

Starting with the code of **diskInfo.py**, you can create a different version of the file that can tell you whether a given disk partition is mounted or not in case you have problems with a hard disk and want to look into it even more.

Network interfaces

The example code for this section, called **netPS.py**, will illustrate how to use `psutil` to get network-related information. The `psutil.net_connections()` method returns a list of network connections depending on the parameter passed to it. You can either choose to use no parameters at all or pass one of the following parameters to it: "inet", "inet4", "inet6", "tcp", "tcp4", "tcp6", "udp", "udp4", "udp6", "unix" or "all". If you're not sure about the kind of network traffic you want, you should pass the "all" argument to `psutil.net_connections()`.

Executing **netPS.py** using root privileges generates the following kind of output:

```
[sconn(fd=1, family=<AddressFamily.AF_UNIX: 1>, type=1,
laddr='private/error', raddr=None, status='NONE', pid=None),
...]
```

If you try to execute **netPS.py** without root privileges, you might get the following kind of error message depending on your Linux setup and the privileges of the user account used for executing `netPS.py`:

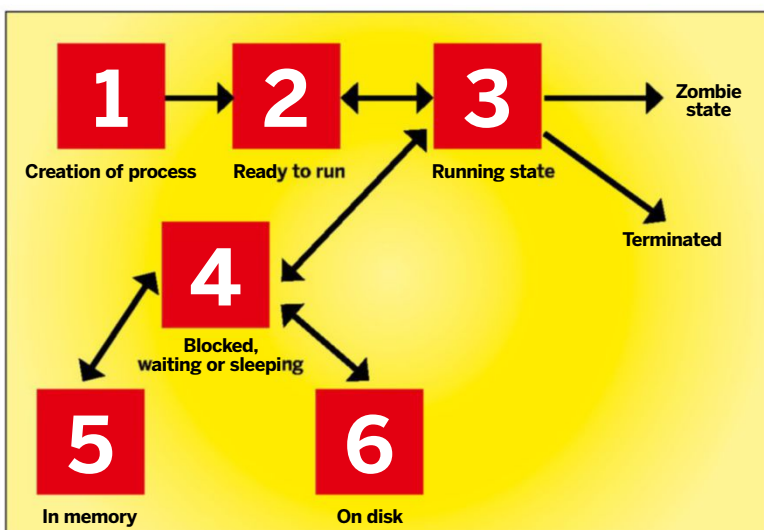
```
$ ./netPS.py
...
psutil.AccessDenied: psutil.AccessDenied (pid=84595)
```

(See the previous page for detailed output from **netPS.py**.)

Quick tip

Should you wish to learn more about the internals of Linux and Unix, you can read *Advanced Programming in the UNIX Environment* by W. Richard Stevens and Stephen A. Rago or *The Design of the UNIX Operating System* by Maurice J. Bach.

» The diagram shows the various process states of a Linux system as well as the way that it goes from one state to another.



The more output that you get from **netPS.py**, the more network connections your machine has. If your Linux machine is a web server then having lots of connections is perfectly acceptable; however, if you have a standalone client, then too many network connections is not likely to be a good thing and might flag up the fact that you need to check the security of your machine.

Logged in users

Let's go on and now learn how to use `psutil` to display information about the logged in users of your Linux machine. The code of **who.py** is the following:

```
#!/usr/bin/env python3
import psutil
for user in psutil.users():
    print(user.name, "at", user.terminal)
You just have to iterate over the return value of the psutil.
users() which does all the dirty work. Executing who.py
generates the following kind of informative output:
$ ./who.py
mtsouk at pts/0
```

As you will likely note, you can also see the terminal each user is connected to.

Examining Apache processes

Each process has a parent process that started it, which is a very common way of starting multiple instances of the same process. A very popular example is the Apache web server: when you start Apache, an apache process is started. This process automatically starts additional apache processes that are used for serving HTTP requests—the number of the apache processes that will be created is defined in the apache configuration file. The key point here is that without knowing the way Apache works, you can't implement what you want to do, much in the same way as understanding `psutil` output.

In this section, we will show you how to use `psutil`, and some command line utilities, to find out the process id of the parent process that started all Apache children processes. The name of the Python script is **apacheParent.py** whereas the name of the Apache process is usually `apache2`. If you aren't sure about the name of the Apache 2 executable, you can find it out pretty easily:

```
$ ps ax | grep -i apache | grep -v grep | awk {'print $5'} | uniq
/usr/sbin/apache2
```

The Python 3 script, called **apacheParent.py**, doesn't require the use of the full path for the Apache 2 executable.

The approach that **apacheParent.py** uses is pretty simple, yet very effective: after finding out all processes with a given name, which is passed as a command-line argument to `psutil`, it finds out the process ID of the parent process of each one of them and prints it. In order to avoid dealing with lists and processing multiple variables in Python, which is possible but time-consuming, the rest of the processing will be done externally using traditional Unix command line tools. If there's not a unique parent process ID, it means that there's something wrong with your Apache 2 server process or that you are running multiple Apache 2 instances that listen to multiple TCP/IP ports.

The important code of **apacheParent.py** is the following:

```
for proc in psutil.process_iter():
    if proc.name() == nameP:
```

» **Want even more coding?** Grab a bookazine at <http://bit.ly/LXFspecial>


```
code$ ./apacheParent.py apache2 | sort | uniq -c
1 1
10 27111
code$ ./apacheParent.py apache2 | sort | uniq -c | sort -rn
10 27111
1 1
code$ ./apacheParent.py sshd | sort | uniq -c | sort -rn
1 3353
1 26847
1 1
code$ ./apacheParent.py init | sort | uniq -c | sort -rn
code$ ./apacheParent.py sshd | sort | uniq -c | sort -rn
2 3353
1 30756
1 26847
1 1
code$ ./apacheParent.py agetty | sort | uniq -c | sort -rn
2 1
code$ ./apacheParent.py mysql | sort | uniq -c | sort -rn
code$ ./apacheParent.py mysqld | sort | uniq -c | sort -rn
1 2141
code$ ./apacheParent.py mysqld
```

► This screenshot shows the **apacheParent.py** script in action identifying the process ID of the parent process.

```
p = psutil.Process(proc.pid)
print(p.parent().pid)
```

As you can see, by combining the capabilities of psutil with other Linux tools and using a bit of imagination you can help you develop great things.

An important warning: please don't assume that the Apache 2 process with the smallest process ID will be the parent process of the other Apache 2 processes. The process ID values are being reset when their reach their maximum value and although you can change this maximum value, it's usually pretty low. You can find the maximum process ID value of Linux machine as follows:

```
$ cat /proc/sys/kernel/pid_max
32768
```

In order to execute **./apacheParent.py** and get the results you want, you should do the following:

```
$ ./apacheParent.py apache2 | sort | uniq -c | sort -rn
10 27111
1 1
```

The first line tells us that the process ID of the parent process is **27111**. The second line of the output indicates that the first Apache 2 process got started by **/sbin/init**, which has a process ID of **1**. The first column shows the number of times a process ID appears in the output, therefore indicating the total number of apache2 processes running on the Linux machine.

It's worth bearing in mind that **apacheParent.py** also works with other programs that start many processes because the name of the process you are looking for is given as a command line argument to it, which can be anything that you want! (See *apacheParent.py* in action, above.)

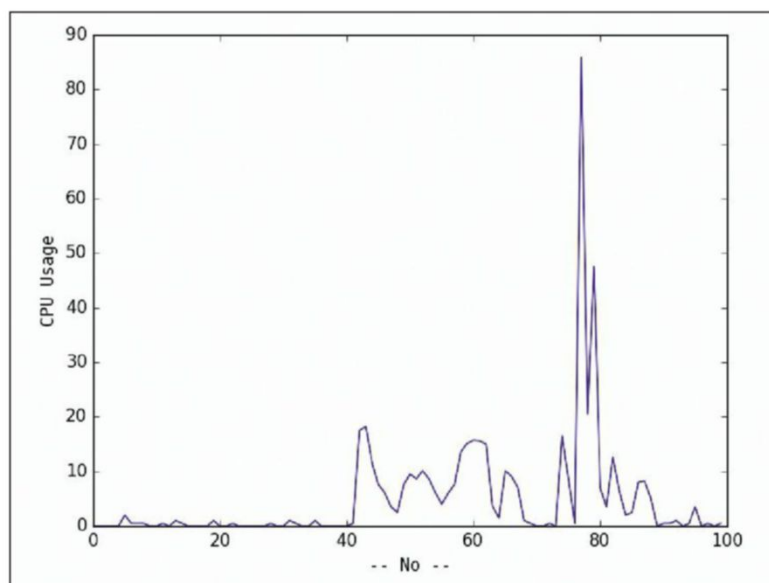
Plotting psutil data

The main benefit you get from using psutil is that you can easily combine the capabilities of Python 3 and its modules with the psutil module. In this last section we will illustrate how to get the output of psutil and create a plot using matplotlib (or Octave, the open source alternative, see Tutorials, p76, **LXF209**) which is a powerful 2D plotting program! The important Python 3 code of **CPUgraph.py** is the following:

```
for k in range(N):
    x.append(k)
    temp = psutil.cpu_percent(interval=2)
    y.append(temp)
```

As you can see, you use the **psutil.cpu_percent()** method to get the current CPU usage, which you add to a list variable that's named **y**. The number of the elements on the list is defined by the value you put in the **range()** function and is given as a command line argument to the program. Now you must wait until after you have collected all the desired data in order to plot it and save the output as a PNG file. Executing **CPUgraph.py** generates the reasonably pretty output (like the example graph that you can see below).

Hopefully, you should now be persuaded by how useful the psutil module can be for advanced Linux users but especially for system administrators. Additionally, Python code that uses psutil can run as a cron job without any changes. As a result, adding psutil to your arsenal of tools will be very beneficial for you. **LXF**



► Here's the graphical output of the **CPUgraph.py** script, which draws the output of **psutil.cpu_percent()** method using **matplotlib** (or you could use **Octave**).

The various states of a process

When Unix was first introduced, computers were less capable as they only had a single CPU and a small amount of RAM. In order for Unix to be multi-user and multi-tasking, it had to be able to run each individual process occasionally, which means that a process should have multiple states, which is what you get by looking at the value of **psutil.Process(PID).status()**.

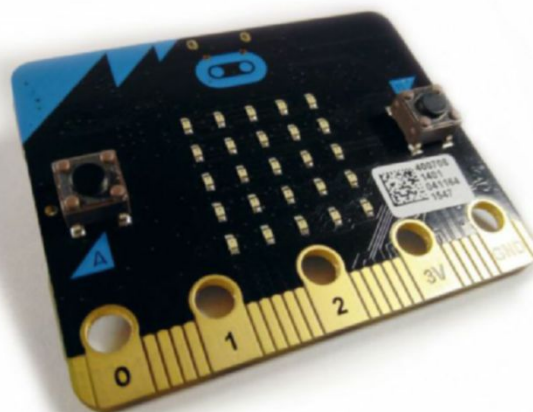
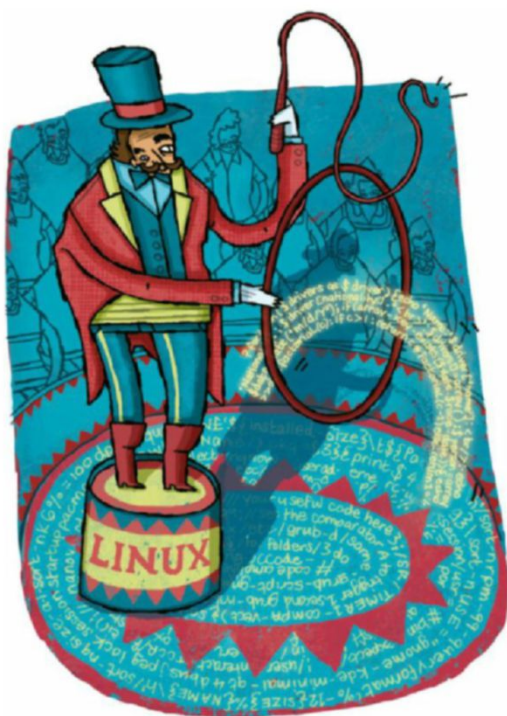
So, a process can be ready-to-run, running, waiting, blocked, sleeping, terminated or in a zombie state (see diagram on left). The last state is rare and shows that there was something wrong with the way the process was terminated. Additionally, when a process is in waiting, blocked or sleeping state, which means that it's not running for some reason, it can reside either

in memory or on disk, which mainly happens due to the limited amount of RAM that machines have. It's important to remember that you can't control the state of a process as this is the job of the Linux scheduler that runs in the kernel. (See the diagram to the left for various states of a Linux process and the routes from one state to another.)

MicroPython:

Light glove

Les Pounder introduces us to MicroPython and uses it with a micro:bit to build a gesture-controlled light glove.



» The BBC micro:bit is a small and unassuming device, but this tiny board can power many different projects from simple lights to powerful robots.

It makes sense to use the micro:bit for this tutorial as it's the most accessible way to start with MicroPython as the board comes with plenty of supporting documentation and projects to gently introduce the platform.

The micro:bit hardware

Measuring just 5cm by 4cm it's a small board, but it's packed with components, such as sensors in the form of an accelerometer and compass. There's a 5x5 single-colour LED matrix on which we can display text or pictures and two push button inputs along with five input/output rings that can be used with crocodile clips to connect to other components. There is also a 20-pin edge connector that will break out all of the IO ports when used with an adaptor. Powering the micro:bit is a 32-bit ARM Cortex M0 CPU with built-in Bluetooth Low Energy (BLE). This CPU is nowhere near as powerful as a Pi, and it's not meant to be, but it provides enough power for the platform.

In this tutorial, the first of a two-part look at MicroPython, we'll get to know the hardware and software and use it to create an interactive wearable light glove that will react to the movements of the wearer thanks to the accelerometer built into the micro:bit. We'll also demonstrate how to use MicroPython by creating projects that are flashed to the micro:bit and can be used again and again, even without a computer connected. We shall also introduce how to use MicroPython interactively with the micro:bit hardware, enabling ideas and logic to be tried out instantly.

MicroPython might sound like programming Python in a tiny font, but it is, in fact, a leaner implementation of Python 3 that's been optimised for use on microcontrollers. It was originally created by Damien George in 2013 as a crowd-sourced project for funding a language development, hardware testing and development platform, which is now known as the pyboard. The original funding target was smashed many times over and the project grew to incorporate other devices, such as the popular ESP8266 – the low-cost Wi-Fi chip with full TCP/IP stack and MCU (Micro Controller Unit) – which is now fully compatible with MicroPython.

We've also seen MicroPython make it onto the micro:bit. Early in the development of the BBC micro:bit project, many programming languages were suggested, but Python was considered the best option largely thanks to the growing trend for using it in education alongside the Raspberry Pi. The Python Software Foundation was contacted and because of its involvement, lead by Nicholas Tollervey with development from Damien George and many members of the Python community, it's now on micro:bit.

Quick tip

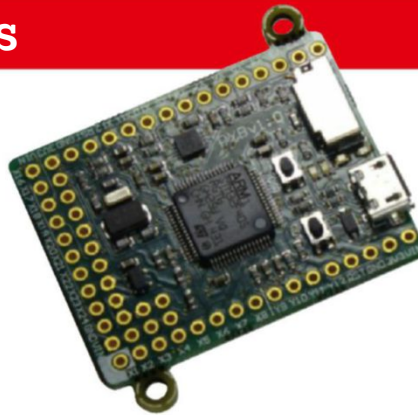
All of the code for this project can be downloaded from GitHub as an archive – <https://github.com/lesp/LXF-MicroPython/archive/master.zip>. We have also included high resolution circuit diagrams of the connections made in the projects.

MicroPython Boards

In this tutorial, we focused on the BBC micro:bit for our introduction to MicroPython. But this is not the only board to support the language, rather there are many more alternatives:

» **The pyboard** The original board developed in 2013 by Damien George is a smaller development board that shares some similarities to the micro:bit, such as a 3-axis accelerometer. It also uses a STM32F405RG micro controller, Cortex M4 CPU that's clocked at 168MHz, and has 192KiB of RAM for your projects. But what if your project is bigger? Fear not as the pyboard comes with a microSD card slot! Additionally, the pyboard comes with 29 IO pins.

There's no dedicated MicroPython editor for this board, rather you plug the board into your



computer and use a standard text editor to write your code. The version of MicroPython used with the pyboard is a little more complex than that

which is supported by micro:bit, but it still follows the standards that are required for it to be considered a Python language. You can read more about it at <https://micropython.org>.

» **Wipy** This is a MicroPython-powered board for Internet of Things (IoT) applications. This board comes with a Cortex M4 processor and a Texas Instruments CC3200 MCU (Micro Controller Unit) for Wi-Fi connectivity. The Wipy board also features 25 IO pins, a real time clock and hash/encryption engines for SHA, MD5, DES and AES, which makes it a good board for IoT research projects. Programming the Wipy is done via a special editor called *Pymakr*. You can read more about it at <https://www.pycom.io/solutions/py-boards/wipy1>.

For this project, you'll need quite a few bits and bobs including soldering equipment. The list includes: a USB battery pack and USB-to-micro USB lead, an 8-24 pixel WS2812B 'neopixel' ring, two crocodile clips, one LED (any colour), five M4 countersunk machine screws that are 12mm length, five nuts and washers for the screws and hookup wire (solid core).

But our first job is to get our system ready to use MicroPython, by installing the simple code editor *Mu*, which is designed for beginner programmers. Installing this is trivial, all you need to do is download the application from <http://bit.ly/Mu4microbit> then navigate to your **Downloads** folder. Right-click on the application and select Properties then go to Permissions and change the permissions so that the file can be executed as an application. Now you can double-click on the application and it will open the *Mu* editor. *Mu* has been designed for those new to coding to just get hands on and code as quickly as possible. The editor is rather sparse compared to others but it focuses on the key features: loading and saving files, flashing code onto your micro:bit and something called REPL which we shall investigate later. Essentially, *Mu* isn't about confusing the user with pointless icons and menus. *Mu* is your first steps into MicroPython, and it's there to help you, e.g. both Micro Python and Python are whitespace sensitive, which means they use indentation to identify code that's inside a loop or a function etc. In some editors the user is forced to add that indentation manually, but *Mu* automatically indents your code for you. The same is true with auto completion; by typing the first few letters of an instruction, e.g. the name of an I/O pin, we see a drop-down list of all potential matches, which is very handy when writing code involving sensors and other inputs. So let's get started with *Mu*.

Are the lights on?

Our first test is to light up an LED, which we shall attach to pin 0. Using the crocodile clips, connect pin 0 of the micro:bit to the long leg of the LED. Then connect the micro:bit's GND pin to the short leg of the LED. Now connect your micro:bit to your computer using a USB lead. In the *Mu* editor we'll write a few lines of code to blink our LED. We start this code by importing the microbit library that's used to enable access to

the micro:bit and its many components: `from microbit import *`. (Note: All the code we've used can be downloaded here:

<https://github.com/lesp/LXF-MicroPython/archive/master.zip>.) Next, we create an infinite loop that will

continuously run the code: `while True:`. In order to turn on the LED we need to supply it with power and to do that we'll turn pin 0 on with:

```
pin0.write_digital(1)
```

By using `write_digital` we can turn a pin on or off, just like a switch. We then use `sleep` to create a delay between turning the LED on and the next step which is to turn the LED off:

```
sleep(1000)
```

If this `sleep` were not between them, then the change would happen instantly and we wouldn't see it.

The way `sleep` is defined in Micro Python is different to Python in that `sleep(1000)` means to sleep for 1 second. Whereas typical Python code uses `sleep(1)` for the same effect. We now write the code that will turn Pin 0 off, which turns off the power to the LED:

```
pin0.write_digital(0)
```

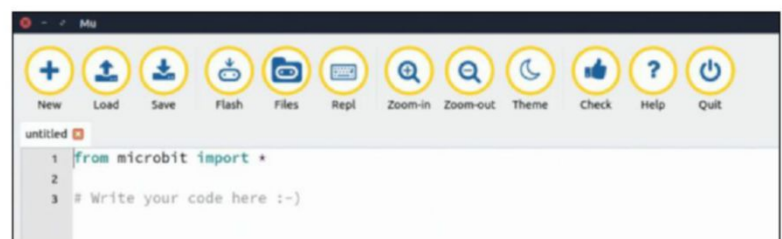
Our final line of code is another sleep to create the desired blink effect:

```
sleep(1000)
```

Now click 'Save' and name the file **blink.py** followed by clicking on 'Flash' to upload the code to your micro:bit. In a few seconds, you will see your LED blink into life.

REPLing down code

REPL is short for Read-Eval-Print Loop and is also known as a shell. In REPL, we can enter commands and algorithms etc and the computer will evaluate the input and print the correct »



» **Mu is a simple editor which focuses on getting beginners create cool projects with a clear user interface and plenty of support.**

» **Improve your code skills** Subscribe now at <http://bit.ly/LinuxFormat>



» response. How this works for MicroPython is that we can have a direct 'conversation' with a connected micro:bit, issue commands and control the board in real time. Using REPL is remarkably simple with *Mu*. To start, all we need to do is ensure that our micro:bit is connected to our computer. Then click on REPL and you'll see the bottom section of the *Mu* editor change and display a console interface, typically with '>>>' to indicate that it's ready to be used. If there are any garbage characters on screen, click twice on the REPL icon to reload.

Using the same circuit as we created in our previous test, let's turn the LED at Pin 0 on. In REPL type the following: `pin0.write_digital(1)` remember to press Enter to run the command. Your LED will now come to life.

We now turn power on to a pin and check the state of a pin. In this case, we'll supply power to pin 0 to do this, which can be used with external inputs, such as switches or buttons. For this test remove the crocodile clip from the long leg of the LED and attach it to the 3V pin of your micro:bit. The 3V pin and 0 should now be directly connected with a single clip. In the REPL type the following to print the state of pin 0: `pin0.read_digital()`. This will report "1", short for "True", which means that pin 0 has power, so our imaginary switch/button has been tripped.

Speaking of buttons, the micro:bit has two, so let's use one. Buttons are marked A and B, and we can check the state of a button by checking if it's currently pressed. If the button is held down, then REPL will respond "True", if not then we see "False":

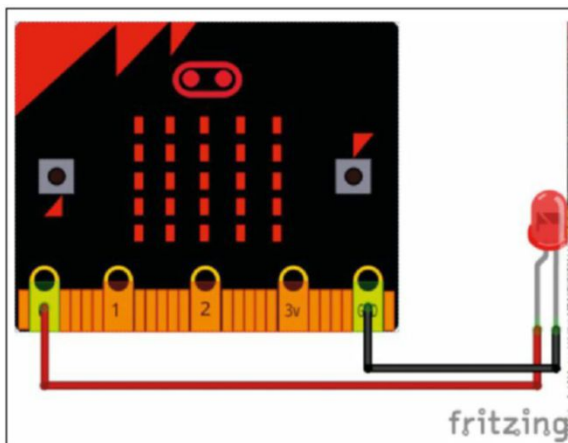
```
button_a.is_pressed()
```

We can also check to see how many times a button has been pressed since the micro:bit was turned on with `button_a.get_presses()`. This should return a low number. Now press the A button lots of times and repeat the previous line of code and you will see the number has increased.

Now for our last REPL exercise, we'll try using the 5x5 LED matrix, which can be used to display scrolling text with, for example `display.scroll("LXF ROOLZ")`. Alternatively, we can use REPL to show simple images, such as a smiling face with `display.show(Image.HAPPY)`, a cute little rabbit `display.show(Image.RABBIT)` or a less cuddly skull `display.show(Image.SKULL)`.

With our walkthrough of REPL complete, click on the REPL icon to close the shell. We now move on to our project. You'll need to remove any crocodile clips and components from the micro:bit in preparation for the project.

» Our simple test circuit is the first to be made with our micro:bit. We're using it to check that our hardware and software is working correctly before proceeding to take on more complex projects.



To consolidate our knowledge of Micro Python for the micro:bit we'll create a motion-controlled light ring that will use accelerometer data for three axis x, y and z. This data is taken from the micro:bit's built-in accelerometer and then used to create a mix of colours using a red, green and blue colour mix on our light ring. Officially called a WS2812B ring, but it's commonly called a 'Neopixel ring' with Neopixel being the brand name.

Running rings

Our WS2812B ring requires a little soldering. We need to solder three wires: one to each of the IN, VCC and GND pins. Give yourself plenty of spare wire as we'll need to connect it to your micro:bit. Strip 2cm of the sleeving from the wires so that bare wire is shown. If you can't solder, then now is a great time to learn, pop along to your local Hackspace or Makerspace and someone will show you how.

The wires from our WS2812B ring need to be connected to our micro:bit. The best way to do this is by using the M4 machine screw. Slide a screw through pin 0 and, with your finger, hold the screw in place and flip the board over. Now slide a nylon washer over the screw before winding the bare wire from our WS2812B's In connection around the screw. At this point, try to keep the wire tight as you wind. Next, you'll need to take an M4 nylon nut and use it to squash the wire between the nut and the washer, ensuring that the wire is not touching anything else. Your wire should be held firmly in place, but don't overtighten as it may damage the board. Repeat this process for the 3V pin and GND pin. Now plug in your micro:bit to your computer running *Mu*.

It's now time to start the code for this project by creating a new file in *Mu*. As ever in Python we need to import some extra libraries:

```
from microbit import *
import neopixel
```

First, we import all of the `microbit` library, giving us access to all of the components present on the board. Next, we import the `neopixel` library, which is used to work with our WS2812B ring. Now we need to create an object that will store the pin used to connect the WS2812B ring to our micro:bit.

```
np = neopixel.NeoPixel(pin0, 24)
```

In this case, `pin0` is connected to the WS2812B In pin. We also need to tell the code that we have 24 pixels in our ring. If you are using a different-sized WS2812B ring then count the pixels and replace `24` with your value, but 24 is really the maximum safe limit.

We now enter into the main loop for the project, this loop will run its contents until the power is turned off.

```
while True:
```

Inside of the loop, all of our code needs to be indented by four spaces automatically. Our first task is to get the raw data for our x axis:

```
reading_x = accelerometer.get_x()
```

To do this we use `accelerometer.get_x()` function which will return a value between -1024 and 1024 and store it inside a variable.

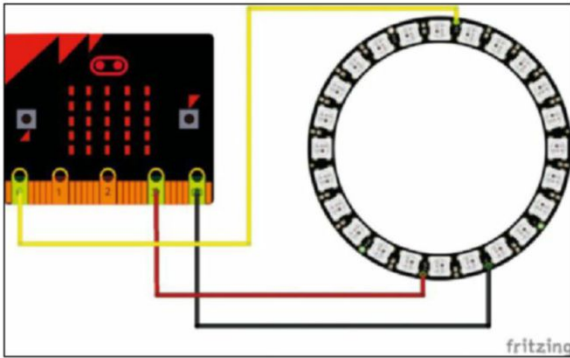
Next, we use an if condition:

```
if reading_x < 0:
```

```
    reading_x =
```

We do this to check that the value of variable `reading_x` and if it is less than zero, it will change the value to zero.

» **Want even more coding?** Grab a bookazine at <http://bit.ly/LXFspecial>



► Our WS2812B ring only requires three connections: 3V power, a Ground connection and a data connection to Pin 0. This pin sends data to the ring to control the pixel colours.

We shall reveal why we do this later in the code. In order to use `reading_x` with our WS2812B ring, we need to convert the raw reading into something more palatable.

The accelerometer data can range from -1024 to 1024. But the WS2812B pixels only work between values of 0 and 255, so if we give them a value greater than 255 this will cause an error. So we need to divide the value of the variable by 10, giving us a maximum value of 102.4, but this is a float value and the neopixel function only works with integers, so we convert the answer into an integer before storing it in the variable. Now you might be thinking "Why divide by 10, it gives a really low number?" True it does, but it also means that we are not pushing the pixels too hard, which means we have a longer battery life, and a less stressed micro:bit:

```
reading_x = int(reading_x / 10)
```

We now repeat these steps for the y and z axis:

```
reading_y = accelerometer.get_y()
```

```
reading_y = int(reading_y / 10)
```

```
if reading_y < 0:
```

```
    reading_y = 0
```

```
reading_z = accelerometer.get_z()
```

```
reading_z = int(reading_z / 10)
```

```
if reading_z < 0:
```

```
    reading_z = 0
```

To ensure that we can see the axis data, we print the values of the three variables to the Python shell, which is accessible via the REPL console while the micro:bit is connected to your computer.

```
print(reading_x, reading_y, reading_z)
```

In order to control the WS2812B ring we need to use a for loop, still inside of our while True loop. This for loop has a

range, which controls how many times it iterates:

```
for i in range(24):
```

In this case, we've set it so that it iterates once for each pixel in the ring, so 24 times.

Coding our colours

Inside of the for loop we create three new variables, red, green and blue. In each variable, we will store the values of our x,y and z axis data, which we collected and converted earlier.

```
red = reading_x
```

```
green = reading_y
```

```
blue = reading_z
```

We now refer back to the `np` object that we created earlier in this tutorial.

```
np[i] = (red, green, blue)
```

```
np.show()
```

This object takes a positional argument, the pixel that we wish to change and as you can see that value is `[i]`. In other words, it will iterate the value each time the for loop iterates, starting with 0 and ending at 23. The object also requires the colour data for the pixel, which we pass as a tuple containing the red, green and blue data. Last, in order to see the change of colour we need to tell the neopixels to `show` the updated colour data.

With our code complete, now is the time to flash the code onto our micro:bit. Click on 'Flash' to upload the code, it should take about 30 seconds to complete. Once ready your micro:bit will reboot and the WS2812B ring will come to life. Now unplug your micro:bit from the computer and attach a USB battery using a USB to micro USB cable. You can also use elastic bands to secure the battery and WS2812B to your arm. Now is the time to fling your arm around and pretend that you are Iron Man! **LXF**



A growing community

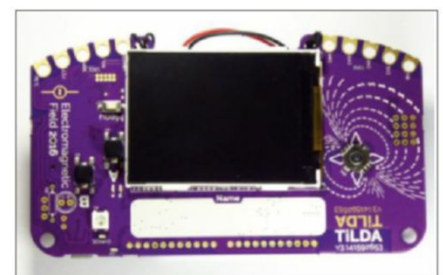
Single board computers and microcontrollers all have one thing in common—they need a community to keep them alive. There are many boards on offer, some with huge communities, such as those following the Arduino and Pi.

With MicroPython, there's a strong community of Python enthusiasts, of which there are many across the world, and whose interests also reach into the world of physical computing/hardware hacking. This was most evident at Electromagnetic Fields 2016, also known as EMF Camp. This festival provides a Glastonbury-like event for those interested in maker activities. But one of the most common

activities for the festival goers is the conference badge. In recent years conference badges have become more than "name holder", rather they come with hackable hardware.

EMF Camp has a tradition of hackable badges and 2016 was no exception. The TiLDA mk3 debuted in 2016 and it was powered by MicroPython. But the TiLDA mk3 came with extras such as a colour LCD screen, accelerometer, Wi-Fi, micro SD storage, an app store and crocodile clip connections. This badge was the result of months of work by a dedicated MicroPython hardware hacking community and is a reflection of what can be achieved. You too

can be part of the MicroPython community, all you need to do is head over to <http://forum.micropython.org> and sign up.



Answers

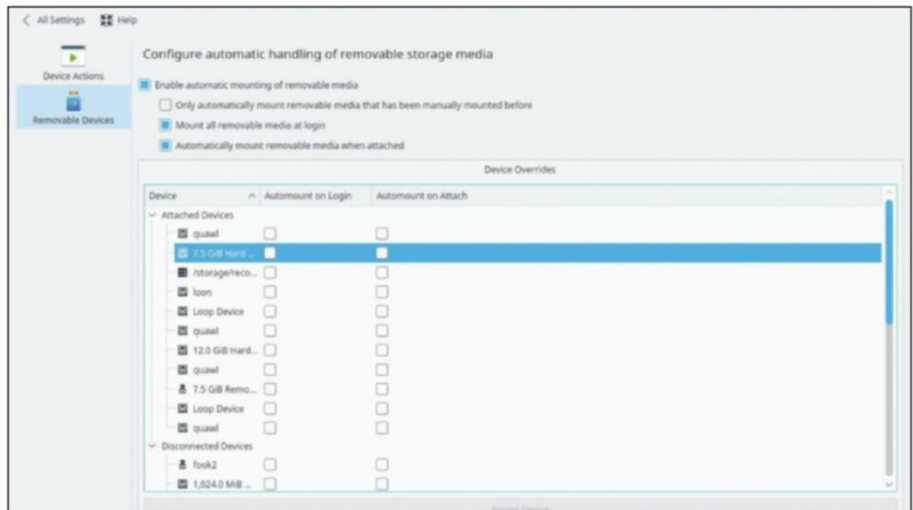
Got a question about open source? Whatever your level, email it to lxf.answers@futurenet.com for a solution.

This month we answer questions on:

- 1 Controlling mount points
- 2 Manipulating the PATH
- 3 Resetting passwords
- 4 Reading a disk image
- 5 Running KDE apps on GNOME
- ★ Reading the *systemd* journal

1 Moving mount points

My desktop distro is Gentoo Linux, which began from a *Linux Format* DVD! Last year, I decided it was time for a new box. My system is on a SSD and /home is on a Seagate Barracuda with a media adaptor and a hotswap rack. It all works well, with one exception: the hotswap rack drive I currently have is automounted at /media. It's a 1TB drive with two partitions. I'd like this to mount at a location on / I call /hotswap rather than /media. Perhaps *LXF* has a suggestion on how I can integrate



Automounters give some control over mounting media, but /etc/fstab should take priority.

these changeable drives. Changes to *fstab* have no impact.

Rob Russell

You don't specify what changes that you have made to */etc/fstab* nor which desktop and

automounter you are using, but */etc/fstab* usually overrides any automounter defaults.

One thing that you need to be careful of dealing with hotswap drives is that the drive name isn't guaranteed to be consistent. Depending on when you connect it and the order in which drives are discovered at boot, it could have a different device node. The solution is to use an identifier that doesn't change. You can use a filesystem's UUID, which you can read with *blkid*, for example: `$ sudo blkid /dev/sdb1`.

Once you have the UUID, you can add an entry to */etc/fstab* like this:

```
UUID=xxxx /hotswap auto defaults 0 0
```

The *auto* in the third field tells the *mount* command to automatically determine the type of filesystem. If you always use the same filesystem on your drives, you can specify it here. If you use different drives in the rack, you will need an *fstab* entry for each one.

The other option is to use filesystem labels. These aren't assigned by default, unlike UUIDs, so you will need to set them up, but they have the advantage of being readable. The command to use depends on the filesystem, for ext filesystems use *e2label*:

```
$ sudo e2label /dev/sdb1  
$ sudo e2label /dev/sdb1 data
```

The first command reports the filesystem's label, if any, the second sets it to data. Then you can specify the label in *fstab* like this:

```
LABEL=data /hotswap ext4 defaults 0 0
```

There are similar commands to work with the labels for other filesystems. If you're

Enter our competition

Win!



Linux Format is proud to produce the biggest and best magazine that

we can. A rough word count of *LXF*217 showed it had 65,010 words. That's a few thousand more than *Animal Farm* and Kafka's *The Metamorphosis* combined, but with way more Linux, coding and free software (but hopefully less bugs). That's as much as the competition, and as for the best, well... that's a subjective claim, but we do sell

way more copies than any other Linux mag in the UK. As we like giving things to our readers, each issue the Star Question will win a copy or two of our amazing Guru Guides or Made Simple books – discover the full range at: <http://bit.ly/LXFspecials>.

For a chance to win, email a question to lxf.answers@futurenet.com, or post it at www.linuxformat.com/forums to seek help from our very lively community.

See page 94 for our star question.



Terminals and superusers

We often give a solution as commands to type in a terminal. While it is usually possible to do the same with a distro's graphical tools, the differences between these mean that such solutions are very specific. The terminal commands are more flexible and, most importantly, can be used with all distributions.

System configuration commands often have to be run as the superuser, often called root. There are two main ways of doing this depending on your distro. Many, especially Ubuntu and its derivatives, prefix the command with `sudo`, which asks for the user password and sets up root privileges for the duration of the command only. Other distros use `su`, which requires the root password and gives full root access until you type `logout`. If your distro uses `su`, run this once and then run any given commands without the preceding `sudo`.

automounter continues to ignore `fstab`, you should investigate the options for the particular automounter you use. For the KDE desktop you will find this in System Settings, while in GNOME it is configured in *Nautilus*.

2 One True PATH

Q I need to permanently add `/usr/local/xds/bin` to the PATH. Currently, there is an existing PATH in `.bash_profile`, like this:

```
PATH=$PATH:$HOME/bin
export PATH
```

However, I'm not sure exactly where and how to add the extra path.

Dave Pritchard

A The PATH variable contains the directories searched for commands when they are given without an explicit path. By default this is only `/bin` and `/usr/bin` for a normal user, the superuser also has `/sbin` and `/usr/sbin` included. You can see the current search path by running:

```
$ echo $PATH
```

The PATH variable contains a list of directories that are separated by colons and you can modify this in the same way as any other environment variable. You may find PATH

already contains more than the two default paths, some may have been added in `/etc/profile` while others in your local profile file; `.bash_profile` for the bash shell.

Setting a variable replaces the contents with the new information, so it's usual to include the existing path in the assignment in order to add to it. The line in `.bash_profile`

```
PATH=$PATH:$HOME/bin
```

does just this. The shell expands the original PATH variable first and substitutes its contents in its place. So if PATH previously contained `/usr/bin:/bin`, the actual assignment being run would be:

```
PATH=/usr/bin:/bin:/home/dave/bin
```

See how both the `$PATH` and `$HOME` variables are replaced with their contents. It is normal practice to have a `~/bin` directory in your path, where users can store their own scripts. You could modify this line to include the path you want or you could add a separate line to make things clearer:

```
PATH=$PATH:/usr/local/xds/bin.
```

You will need to put this line before the `export PATH` line. The `export` command makes the modified PATH available to the shell and any commands that it runs, otherwise it would be local to `.bash_profile`. Doing this will only add the path for your user, if you want to make this path available to all system users, you will need to put in either `/etc/profile` or a new file in `/etc/profile.d`. The latter is a better choice if your particular distro uses it as it will not be overwritten by an update, while `/etc/profile` may.

The order of the paths is important, the paths are searched from left to right. If the existing `.bash_profile` line reads: `PATH=$HOME/bin:$PATH` your `~/bin` directory would be searched first, so if an intruder placed a script in here with the same name as a system command that script would be run first. That's why it's safer to add new paths to the end. If you really want to override the behaviour of an existing command, consider using an alias instead.

3 Password reset

Q I have acquired a Dell laptop with Linux Mint installed. I tried to update the OS and install software but it asks me for a password, which I don't have. How do I find the password?

Ed Scott

A Linux Mint is a derivative of Ubuntu Linux and uses the `sudo` command to run programs that need superuser permissions. This means you use your normal user password, the one you use to log in when you boot. However, that piece of advice isn't going to be much help if the computer was set up to automatically login to the desktop when it starts up. If this is the case, you need to boot from a live CD and reset your user password. Most live CDs will do the job, although we prefer System Rescue Cd (<http://system-rescue-cd.org>). This boots to a console by default, but that is fine as the reset process uses console commands. Before you do this, find the device holding your root partition in Mint. You can do this from Mint by opening a terminal and typing `$ df`. The `df` command actually reports free space, but helpfully also includes the device name as the first item on a line; it will be something like `/dev/sda1`. You also need your username, which you can usually see in the menu bar or you can view in the terminal by running

```
$ echo $USER
```

Now boot into the live CD, open a terminal if not already at a console and run:

```
$ sudo mkdir -p /mnt/mint
```

```
$ sudo mount /dev/sda1 /mnt/mint
```

```
$ sudo chroot /mnt/mint
```

The first two commands create a directory and mount your Mint disk on it. The `chroot` command changes the working environment to that directory. Now you are effectively logged into your Mint system as the root user, so change the password for your user with `$ passwd youruser`. It will prompt you for the new password twice. When all is done, press »



A quick reference to...

sudo

One factor in the security of Linux is that you are usually running as a user without root privileges. However, there are times when you need those privileges, usually for administrative purposes, so you need to switch to the superuser (or root). This is traditionally done with the `su` (switch user) command, but that gives full root access, allowing the user to do anything that root could for as long as they are logged in. It also requires them to have the root password. Wouldn't it be safer if users were only allowed

to run as root for the commands they need? This is what `sudo` does. The configuration file, `/etc/sudoers`, contains a list of commands that each user is allowed to execute with root privileges. Running `sudo somecommand` prompts the user for their password and runs the command. It's the user's password that `sudo` needs, not the root password, you don't need to divulge that to anyone. This, combined with the fact that only specified commands can be run, greatly improves security. This is the way Ubuntu works, which is why you don't need a root password with it.

A typical entry in `/etc/sudoers` may look like:

```
nelz ALL = NOPASSWD: /bin/mount,/bin/
```

`umount`

This lets the user `nelz` run `mount` or `umount` on all hosts. You can remove the need for them to enter a password like this

```
nelz ALL = NOPASSWD: /bin/mount,/bin/umount
```

Use this with care! You can define aliases for groups of users or commands to make administering a complex setup simpler. You should not edit `/etc/sudoers` directly, use `visudo`, as root, to edit it. Contrary to the name, this doesn't necessarily use `vi` to edit the file. The command checks the syntax of your edits before writing them to `/etc/sudoers`, so you don't lock yourself out with a typo.

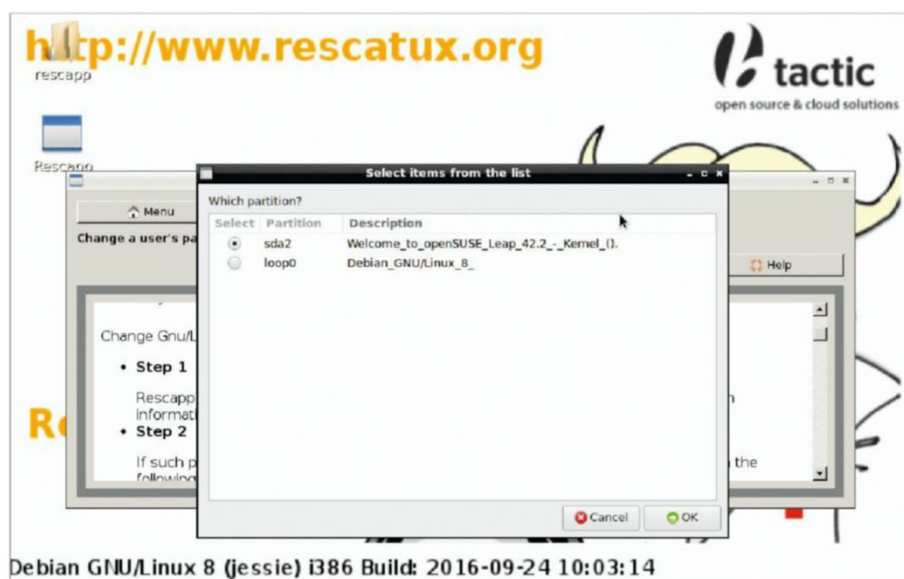
» Ctrl+d to exit the chroot, safely unmount the Mint disk with `$ sudo umount /mnt/mint` and reboot. This procedure should work with just about any live CD, but if you have the Rescatux rescue disc (www.supergrubdisk.org/rescatux) available, there's a graphical alternative. After booting from the Rescatux CD, you will find a button labelled 'Change Gnu/Linux Password'. Press this, read the instructions then follow the steps to reset the password. Rescatux first scans your disc for suitable distros then asks you to pick one—there should be only one in your case. Then it lists all the users on that system, your user should be near the end of the list. Make sure you get the right user, then give the new password when prompted.

Whichever method you use, it's advisable to change the password again once you are booted into the system, just to make sure everything is consistent. To do this, simply open a terminal and run `$ passwd`. It will ask for your current password, which is why we could not do it this way in the first place and will prompt you for a new one.

4 Reading a disk image

Q I've created an image of a hard drive with `dd` and wish to access files in it. I have tried mounting the `image.dd` file but it doesn't work. It also gives an error about `/dev/loop0` already being mounted and I don't even know what that is. Is there a way to get at the files without having to write the whole image back to a disk?

Brian Karlson



» Rescatux makes it easier to reset forgotten Linux passwords.

A Did you image the whole disk or just a partition? If it was only a partition, you can mount this using the loop option with mount:

```
$ mount image.dd /mnt/somewhere -o loop
```

If you created an image of the whole disk this will not work, in the same way, you'll find `mount /dev/sda` doesn't work, even if the disk is all one partition. That's because the disk starts with metadata – like the partition table – so even the first partition isn't at the beginning of the disk. It is possible to use `sfdisk` to determine the offset and pass that to `losetup`

in order to create a suitable loop device that you can then mount. That's a fairly involved puzzle, but fortunately it's one that occurs often enough for someone to come up with a tool to make it simpler.

That tool is called `kpartx` and you may already have it installed, if not most distros make it available in a package. The `kpartx` tool has other uses but here we use it to access partitions on a disk image using:

```
$ kpartx -l image.dd
```

This will list the partitions it finds in the image (actually, running it with no switches

Star
Question

Winner!

This month's winner is Ian Couchman. Get in touch with us to claim your glittering prize!

★ Another journal

Q I'm having a problem with a new install of Debian Jessie on a new laptop. After installing bumblebee-nvidia, the system freezes when I boot (keyboard is dead, can't ssh in). Ah! I thought. Look at the logs. So I booted to another version of Debian on the same machine and looked at `/var/log` on the iffy system. There were no entries in any files I could see with a date stamp corresponding with the attempted boot. I know how to look at the journal for the currently running system, but how do I view the journal from another system? Where is the file?

Ian Couchman

A `Systemd` stores its journal files in a number of locations. Volatile logs are kept in `/run/log/journal` and persistent ones in `/var/log/journal`. As `/run` is a tmpfs, its contents disappear on shutdown or reboot, but it's able to log information before the filesystem containing `/var` is mounted (the

log entries are then synced) so `systemd` can keep logs right from early boot.

There are options to read logs in a different location, but first you have to make sure you have logs. The Storage option in `/etc/systemd/journald.conf` controls where the logs are kept. When set to "volatile" the journal is only saved in `/run`, a setting of "persistent" also saves to `/var/log/journal`, creating the directory if necessary. The default is "auto", which saves to `/var/log/journal` only if that directory exists (which it doesn't on a vanilla install of Debian or Ubuntu). To force `systemd` to save the journal to disk, create the `/var/log/journal` directory and have it owned by the root user and the `:systemd-journal` group. Alternatively, change the Storage setting to "persistent" in `/etc/systemd/journald.conf`. To avoid this being overridden by a subsequent update, it's best to create a file in `/etc/systemd/journald.conf.d`, say `storage.conf`, containing

```
[Journal]
```

Storage=persistent

The default configuration file contains all the settings showing their defaults but commented out. To change them either remove the comment or copy the setting to a file in `/etc/systemd/journald.conf.d`. Once you have `systemd` logging to disk, you can access the journal from another system (providing it runs `systemd`) with `journalctl`. If you mount your Debian root partition at `/mnt/debian`, you can now view the journal with one of:

```
$ journalctl --root /mnt/debian
$ journalctl --directory /mnt/debian/var/log/journal
```

The former is easier, providing your Debian install has `/var` on the root filesystem. If it's mounted elsewhere, use the latter example and give the path to the `log` directory. You can use all of the usual `journalctl` options to examine and filter the information in the journals as if you were working with the current system's journal.

gives the same output, but it's best to be explicit about what you want). To be able to mount them you need to create device nodes in **/dev**, which you do with:

```
$ kpartx -av image.dd
```

The device nodes are created in **/dev/mapper**. The first partition on the first image file will be **/dev/mapper/loop0p1** and so on—you can have more than one image mounted at the same time. You can mount this device node just as you would a normal partition block device with:

```
$ mount /dev/mapper/loop0p1 /mnt/somewhere
```

When you've finished doing what you want with it, unmount the filesystem and then use **kpartx** to release the device nodes:

```
$ umount /mnt/somewhere
$ kpartx -d image.dd
```

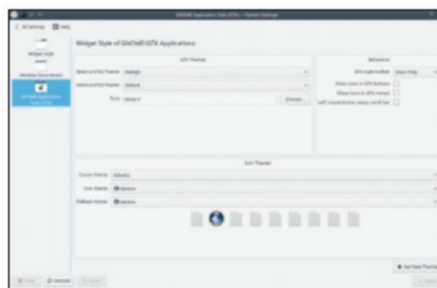
As you are creating device nodes, you need to be root to do this. If you are logged in as a normal user, you'll need to prefix all of these commands with **sudo**.

5 GNOME vs KDE

Q In LXF218 you had an article on *KolorManager* [HotPicks p55]. Can this be used in a GNOME environment? I know there's *colord* for Gnome but I'd like to try *KolorManager* out, but I am not sure if it will run OK.

JohnA1954

A KDE and GNOME are integrated desktop environments, where everything is designed to work together and fit in visually. However, it's quite possible to run GNOME applications on KDE



› **Mixing KDE and GNOME software works but can be a little ugly. However, a careful choice of themes can help.**

and vice versa. There are two drawbacks that spring to mind in doing this. The first is that both GNOME and KDE offload common functions into shared libraries, which saves reinventing the wheel and helps to ensure consistent behaviour between applications. This program, in particular, depends on KDE's System Settings program, so that will be installed by your package manager. This means that the first time you install a KDE program on a GNOME system you will also end up installing quite a few other KDE and Qt packages—conversely installing a GNOME program on KDE will pull in GNOME and GTK packages. This isn't really a problem, the package manager will take care of it and it only really becomes a concern if you are tight on space on the filesystem holding **/usr**.

The other concern is mainly cosmetic: a KDE program running on GNOME (or vice versa) will look out of place. Whether this matters is up to you, we've known to run a few GTK programs on KDE desktops, but we're

Help us to help you

We receive several questions each month that we are unable to answer, because they give insufficient detail about the problem. In order to give the best answers to your questions, we need to know as much as possible.

If you get an error message, please tell us the exact message and precisely what you did to invoke it. If you have a hardware problem, let us know about the hardware. If Linux is already running, use the *Hardinfo* program (<https://github.com/lpereira/hardinfo>) that gives a full report on your hardware and system as an HTML file you can send us.

Alternatively, the output from *lshw* is just as useful (<http://ezix.org/project/wiki/HardwareLiSter>). One or both of these should be in your distro's repositories. If you are unwilling, or unable, to install these, run the following commands in a root terminal and attach the **system.txt** file to your email. This will still be a great help in diagnosing your problem.

```
uname -a >system.txt
lspci >>system.txt
lspci -vv >>system.txt
```

more concerned about how they work than how they look. It's possible to install themes for the various desktops that look like the other, so your applications can be made to blend in. It's up to you whether this is worth the effort and it's not necessary to get programs from a different desktop to run.

One problem that occurs occasionally is programs not showing up in the menus of 'foreign' desktops. This is because the **.desktop** file for a program – the file that controls menu and launcher entries – may contain a line that blocks its display on the menus of other desktops. If this happens and there is no menu entry, simply create a launcher or menu entry yourself. **LXF**



Frequently asked questions...

Wine

› What is Wine?

The name is an acronym of *Wine* Is Not an Emulator.

› So, if it's not an emulator, then what is it?

Well, it is a sort of emulator [NO!–Ed] really. Wine is an open source implementation of the Windows Application Programming Interface (API) on top of X and Unix.

› Does WINE let me run Windows on Linux?

Not exactly. *Wine* is a compatibility layer between Windows programs and Linux. When you run a Windows program with *Wine*, the program thinks it's running on Windows, but the calls it makes to

Windows functions are translated into Linux calls.

› Do I need Windows installed to use Wine?

No, but *Wine* will use Windows DLLs from an existing installation if you have them. If not, it has its own alternatives.

› If I have Windows installed already, why would I need Wine?

First, it means you can run a Windows program on Linux without rebooting into Windows and returning to Linux afterwards.

Wine also makes it possible for you to use Windows plugins in Linux programs, such as web browsers and media players. This gives Linux software programs access to formats that are only handled by proprietary Windows-only codecs.

› Emulators can be slow, how fast is Wine?

Wine only emulates the Windows API, not the hardware. This means programs run at similar speeds under *Wine* as natively on Windows, sometimes even faster. The disadvantage of this is that *Wine* can only run on the same hardware as Windows; it's not a processor emulator.

› How is it available?

You can download *Wine* in a variety of package formats. Packages are also available from most distros' repos. There are also two commercial variants of *Wine*. Codeweavers' *CrossOver Office* is set up to run various Windows productivity applications. It also runs *Internet Explorer* plugins in a number of Linux web browsers, including *Firefox* and *Konqueror*.

All the hard work of configuration is taken care of for you. There are about fifty supported applications listed on Codeweavers' website, but many more work with it.

› That's fine for productivity, what about games?

There was a commercial product called *Cedega*, but was discontinued. The best option for playing Windows games on Linux is plain *Wine*, although for serious Windows gaming, Windows itself is still the best [NO!–Ed] choice.

› Where can I get it?

Wine is in most distros' package repos. You can get the source and packages for many distros from www.winehq.com. *CrossOver Office* is available from www.codeweavers.com, including a free trial version.

On the disc

Distros, apps, games, books, miscellany and more...

The best of the internet, crammed into a phantom-zone like 4GB DVD.

Distros

The good thing about open

source is that anyone can take a piece of software and make any changes they want. The bad thing about open source is that anyone can take a piece of software and make any changes they want! This is why software such as Ubuntu and Firefox have conditions about distributing modified copies. You are free to change the software as you want, but you cannot redistribute your version unless you remove the original branding. This is not about protecting the companies' intellectual property but their reputation. Ubuntu would not be too pleased if we put a broken fork of its distro on the DVD and called it Ubuntu.

So when we produce a desktop remix of Ubuntu, we leave the original ISO unchanged – you can still use Ubuntu as Canonical intended. The remix is a separate file, loaded using Ubuntu's persistence feature, which allows you to save any changes to a disk or file. We had to modify the init script to do this, but it's open source; anyone can change anything! If you are interested in the persistence feature and want to know more, see <https://help.ubuntu.com/community/LiveCD/Persistence>.

Neil

Live distro desktop collection

64-bit

Ubuntu 16.10 Remix

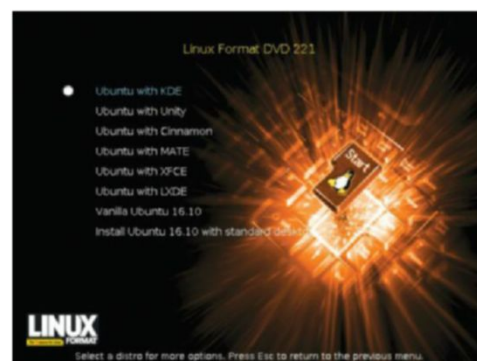
Sorry you missed out last time around, but here is one of our multi-desktop Ubuntu remixes. This one has six key desktop environments for you to try out: KDE, Unity, Cinnamon, MATE, XFCE and LXDE. Yes, we have included other desktops in the past, but this is a way for you to try the main desktops that are supported (either officially or sort of unofficially) by Canonical. You can choose the one you want to try from the boot menu. When you want to try another one, you can log out, select a different desktop from the login screen and log back in (login details below). This can sometimes cause problems when the new desktop tries to start a program that clashes with something the last desktop did. The solution, if that happens, is to reboot and start afresh.

This is a live distro for you to experiment with the various desktops – there is no installation option on the remix desktops. Because of the way the remix is done, installation is not possible – it proceeds most of the way through before failing. It doesn't even have the decency to fail early on! So, we removed the option. Instead, we have included an option to boot from the vanilla Ubuntu ISO image, from where you can install it. Of course, that only gives you the stock Unity desktop but adding one or more of the others is easy. First install *Synaptic* – this is a far more versatile package manager than Ubuntu's own offering. Go into the repositories list and enable the **universe**

and **multiverse** repositories, then hit the refresh button. Now you can install one or more of **kubuntu-desktop**, **cinnamon-desktop-environment**, **mate-desktop-environment**, **xubuntu-desktop** or **lubuntu-desktop**. These are meta-packages that install the whole of the corresponding desktop environment for you.

One advantage of waiting for the remix this time is that quite a few components of the system have been updated since the initial Ubuntu 16.10 release, so you get a more up-to-date and bug-fixed set of desktops to play with.

Login details: username is **ubuntu** and the password is blank, which is to say nothing.



» Ubuntu running the MATE desktop, one of six you can choose from when booting this month's Linux Format DVD. Aren't we lovely?!



» Important NOTICE!

Defective discs

For basic help on running the disc or in the unlikely event of your *Linux Format* coverdisc being in any way defective, please visit our support site at: www.linuxformat.com/dvdsupport. Unfortunately, we are unable to offer advice on using the applications, your hardware or the operating system itself.



New to Linux? Start here

- » What is Linux? How do I install it?
- » Is there an equivalent of *MS Office*?
- » What's this command line all about?
- » How do I install software?

Open [Index.html](#) on the disc to find out



Are you reading this on a tablet?

Download your DVD from www.linuxformat.com/archives

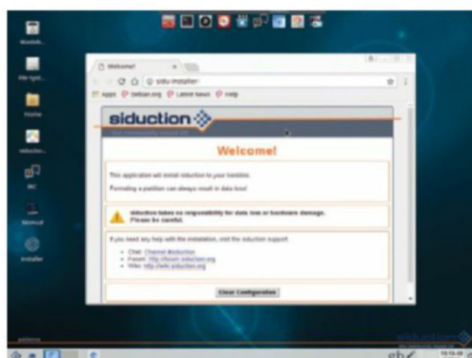
Speedy, lightweight distro

32-bit

Siduction 16.1.0 XFCE

Siduction is a 32-bit distro that uses the lightweight XFCE desktop, making it both fast and pleasant to use. Siduction is based on Debian's unstable branch, which is codenamed Sid. Don't let the word 'unstable' put you off – it doesn't mean it's full of software that is likely to crash. Debian uses 'stable' in the sense of not changing – its stable releases stick with the same software for a long time, so the unstable branch is the more up-to-date software that may make its way into stable one day.

Login details: username **siducer**, password **live**.



Cinnamon-flavoured distro

32-bit

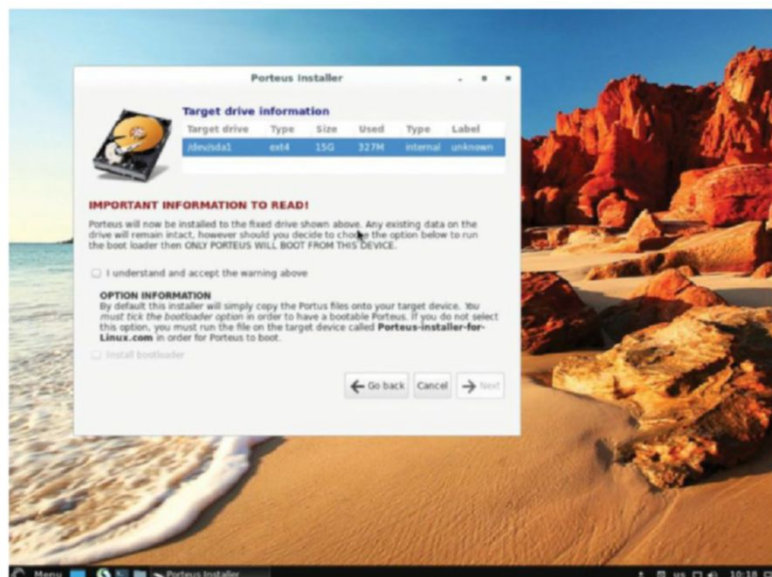
Porteus 3.2.2

I don't think we have featured Porteus before, but this is close to the ultimate in lightweight distros. This is the version with the Cinnamon desktop. Cinnamon is not noted for being especially lightweight but the entire distribution is held in a 273MB ISO image. Maybe that's a bit of a cheat, though, because Porteus doesn't include a web browser as part of its standard software – you need to install it separately, although that is easy

enough with the package manager, and it means you get to choose between *Firefox* and *Chrome*.

Installing Porteus is a little unusual, too. You need to prepare your disk first, using *GParted* (or you can use the option to launch this from within the installer). You must create an empty partition for the **root** filesystem, one for **swap** and, if you want, another for **/home**. You can format the **root** and **home** partitions using whichever

filesystem you prefer – use **ext4** if you're in any doubt – then reboot Porteus and run the installer. If you do not tick the **Install Bootloader** box – which you shouldn't if you want to dual-boot – you need to run the **Porteus_installer-for-Linux** file from the boot directory of your Porteus installation to start Porteus.



And more!

System tools

Essentials

Checkinstall Install tarballs with your package manager.

Coreutils The basic utilities that should exist on every operating system.

HardInfo A system benchmarking tool.

Kernel Source code for the latest stable kernel release, should you need it.

Memtest86+ Check for faulty memory.

Plopp A simple manager for booting OSes, from CD, DVD and USB.

RawWrite Create boot floppy disks under MS-DOS in Windows.

Smart Boot Manager An OS-agnostic manager with an easy-to-use interface.

WvDial Connect with a dial-up modem.

Reading matter

Bookshelf

Advanced Bash-Scripting Guide Go further with shell scripting.

Bash Guide for Beginners Get to grips with *Bash* scripting.

Bourne Shell Scripting Guide Get started with shell scripting.

The Cathedral and the Bazaar Eric S Raymond's classic text explaining the advantages of open development.

The Debian Administrator's Handbook An essential guide for sysadmins.

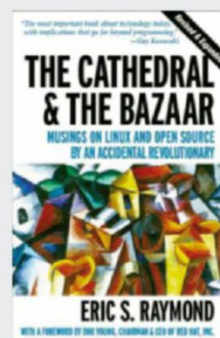
Introduction to Linux A handy guide full of pointers for new Linux users.

Linux Dictionary The A-Z of everything to do with Linux.

Linux Kernel in a Nutshell An introduction to the kernel written by master hacker Greg Kroah-Hartman.

The Linux System Administrator's Guide Take control of your system.

Tools Summary A complete overview of GNU tools.





Master The Terminal

Take control of the ultimate interface to any operating system, the Terminal; work smarter, not harder!

More privacy!

We test out the best privacy distros so you can stay safe and secure while you're online.

Easy file sharing

Not all file sharing is bad sharing, we take a look at the open source file sharing options.

Tiling window managers

There are no three words more sexy in the English language than those—let's do this!

Contents of future issues subject to change – we might never want to leave the enveloping blackness.

LINUX FORMAT

Get into Linux today!

Future Publishing, Quay House,
The Ambury, Bath, BA1 1UA
Tel 01225 442244
Email linuxformat@futurenet.com

EDITORIAL

Editor Neil Mohr
neil.mohr@futurenet.com
Technical editor Jonni Bidwell
jonni.bidwell@futurenet.com
Operations editor Chris Thornett
chris.thornett@futurenet.com
Art editor Elfrain Hernandez-Mendoza
efrain.hernandez-mendoza@futurenet.com
Editorial contributors Neil Bothwick, Jolyon Brown,
Nate Drake, Matthew Hanson, Jody Macgregor, Adam
Oxford, Les Pounder, Afnan Abdul Rehman, Mayank
Sharma, Shashank 'back from the dead' Sharma,
Zak Storey, Alexander Tolstoy, Mihalios Tsoukalos
Cover illustration magictorch.com
Cartoons Shane Collinge

ADVERTISING

Commercial sales director Clare Dove
clare.dove@futurenet.com
Senior advertising manager Lara Jaggon
lara.jaggon@futurenet.com
Advertising manager Michael Pyatt
michael.pyatt@futurenet.com
Director of agency sales Matt Downs
matt.downs@futurenet.com
Ad director – Technology John Burke
john.burke@futurenet.com
Head of strategic partnerships Clare Jonik
clare.jonik@futurenet.com

MARKETING

Marketing manager Richard Stephens
richard.stephens@futurenet.com

PRODUCTION AND DISTRIBUTION

Production controller Nola Cokely
Head of production UK & US Mark Constance
Distributed by Marketforce, 2nd Floor, 5 Churchill
Place, Canary Wharf, London, E14 5HU

LICENSING

Senior Licensing & Syndication Manager Matt Ellis
matt.ellis@futurenet.com Tel +44 (0)1225 442244

CIRCULATION

Trade marketing manager Juliette Winyard
Tel 07551 150 984

SUBSCRIPTIONS & BACK ISSUES

UK reader order line & enquiries 0344 848 2852
Overseas order line & enquiries +44 344 848 2852
Online enquiries www.myfavouritemagazines.co.uk
Email linuxformat@myfavouritemagazines.co.uk

THE MANAGEMENT

Creative director, Magazines Aaron Asadi
Editorial director Paul Newman
Group art director Graham Dalzell
Editor-in-chief, Technology Graham Barlow

LINUX is a trademark of Linus Torvalds. GNU/Linux is abbreviated to Linux throughout for brevity. All other trademarks are the property of their respective owners. Where applicable code printed in this magazine is licensed under the GNU GPL v2 or later. See www.gnu.org/copyleft/gpl.html.

Copyright © 2017 Future Publishing Ltd. No part of this publication may be reproduced without written permission from our publisher. We assume all letters sent – by email, fax or post – are for publication unless otherwise stated, and reserve the right to edit contributions. All contributions to Linux Format are submitted and accepted on the basis of non-exclusive worldwide licence to publish or license others to do so unless otherwise agreed in advance in writing. Linux Format recognises all copyrights in this issue. Where possible, we have acknowledged the copyright holder. Contact us if we haven't credited your copyright and we will always correct any oversight. We cannot be held responsible for mistakes or misprints.

All DVD demos and reader submissions are supplied to us on the assumption they can be incorporated into a future cover-mounted DVD, unless stated to the contrary.

Disclaimer All tips in this magazine are used at your own risk. We accept no liability for any loss of data or damage to your computer, peripherals or software through the use of any tips or advice.

Printed in the UK by William Gibbons on behalf of Future.



Future is an award-winning international media group and leading digital business. We reach more than 57 million international consumers a month and create world-class content and advertising solutions for passionate consumers online, on tablet & smartphone and in print.

Future plc is a public company quoted on the London Stock Exchange (symbol: FUTR).
www.futureplc.com

Chief executive officer Zillah Byng-Thorne
Non-executive chairman Peter Allen
Chief financial officer Penny Ladkin-Brand
Creative director, Magazines Aaron Asadi
Tel +44 (0)1225 442 244

We are committed to only using magazine paper which is derived from well-managed, certified forestry and chlorine free manufacture. Future Publishing and its paper suppliers have been independently certified in accordance with the rules of the FSC (Forest Stewardship Council).





The Electronic Frontier Foundation is the leading nonprofit organization defending civil liberties in the digital world. Founded in 1990, EFF champions user privacy, free expression, and innovation through impact litigation, policy analysis, grassroots activism, and technology development. We work to ensure that rights and freedoms are enhanced and protected as our use of technology grows.

EFF.ORG

ELECTRONIC FRONTIER FOUNDATION

Protecting Rights and Promoting Freedom on the Electronic Frontier

LOSE YOURSELF IN A WORLD OF

Vinyl

FIND YOURSELF IN
OXFAM'S ONLINE SHOP

oxfam.org.uk/shop



OXFAM