DISCOVER ANSIBLE THE NEW PUPPET MASTER

LINUX FORMAT

Get into Linux today!

46 pages of projects + Raspberry Pi

» Create a Pi Tor box
» Code for the kernel
» Go green with Linux

PLUS Make money and run a Linux business

100 BEST LINUX APPS

Your essential guide to all the must-have open source apps

Open NHS

“We exert influence over the entire healthcare community around the use of open source”

Peter Coates on open source in the NHS p44

Beat Big Brother

» The best anonymising distros tested and rated

Guitarix guide

School of rock!

» Pro-level virtual music amps come to Linux

Networking fu

Build a router

» Boost security and hack your own router

Future

TuxRadar.com

LINUX FORMAT
Get into Linux today!

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 GPLv3.

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

Corporate Patron Free Software Foundation

Who we are

This issue we asked out experts: What tool do you find essential, couldn't live without and why? And no, you can't all say *Emacs*...

Jonni Bidwell
Well, I never leave the house without *Chrome*, but that's another story. As far as FOSS goes, for me it's all about the little things so often taken for granted: *grep*, *ping*, *tail*, *lsof* and *nano* (because sometimes *vi* is too hard) can solve so many problems. Also *MPD* controlled by phone is the badger's nadgers.

Neil Bothwick
Well I do use *Emacs*, but… I'm not sure there is anything so essential. There is so much choice in the free and open source world that if anything disappeared I'd be able to find a usable alternative. Having said all that, I'd be hard pushed to do without *ZShell* running in screen session over SSH.

Sean Conway
Mozilla Firefox is the go-to FOSS for me! I cut from Microsoft's SuperMarioNation Internet Explorer software when AOL released *Netscape*. I was a last holdout with that, staying with the web browser until it reached its end of life. A weaning to *Firefox* was the alternative. No amount of *Chrome* will give me cause to change.

Les Pounder
The Arduino project is very dear to my heart. It started the Maker movement with art students in Italy, and thanks to its pricing and ease of use it's become a huge success. Children and adults around the world have built projects great and small thanks to the blue PCB and its great set of resources.

Mayank Sharma
VirtualBox! It's taken the fear out of testing beta software and unfamiliar distros. I can create isolated environments using geekier solutions such as *OpenVZ* and *LXC*, but I still prefer the idiot-proof *VirtualBox* and save all my geekiness for tinkering with whatever it is that I need the virtual environment for.

The best of FOSS

Today, we're celebrating the best of FOSS and championing the 100 best open source software projects. Software that you must have and shouldn't miss out on. From big-names like *LibreOffice* to the tiniest tools such as *rtorrent*, we're running a catalogue of our top 100 must-have tools – and it even includes 70 that will work on the old and new Raspberry Pi!

Of course, if you really love a piece of software, why not contribute to it? That's the real beauty of open source: the ability to get involved with the most exciting or your most loved projects, whether big or small. We're sure we've missed out on some favourite tool or project here, and we're also sure you'll let us know what that is. So make sure you email us and we'll let everyone know Mailserver on page 12.

But what use is software without something to run it on? For once we have three systems that come with Linux pre-installed in our Reviews section starting on page 17. We have a new HP workstation-class laptop, a Dell workstation desktop and the latest Chromebook from Acer. It's certainly a sign of the time that we're able to run regular system reviews with PCs that have a flavour of Linux as a default OS option.

I'm also excited that we're interviewing Peter Coates on how the British NHS is using open source in its infrastructure on page 44. Alongside this a feature on how you can not only run a business with open source, but how a business can deploy and maintain open source systems on page 48. We've got a new sysadmin's coding series from Dr Chris Brown on page 84 while another Brown: Jolyon Brown takes the reins of Administeria on page 56 and Jonni looks into MariaDB on page 88. So enjoy the issue and we'll be back with more next month!

Neil

Neil Mohr Editor
» neil.mohr@futurenet.com

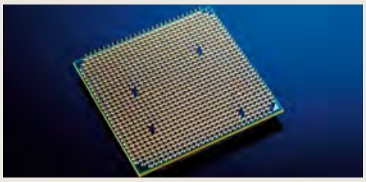
LINUX FORMAT

Contents

"We become what we behold. We shape our tools and then our tools shape us."

Reviews

› **A workstation-class desktop to make Linux fly like it deserves!**



› **If you can spot the eight cores from here, you're better than we are.**

```
FLASHPROXY_PORT=9000

####################################
## Transparent Proxy Settings ##
####################################

## Read first:
## https://www.whonix.org/wiki/Stream_Isolation

##########################
## For the Workstation ##
##########################

## Enable (1) or disable (0) transparent TCP proxying for the Whonix-Worksta
## Any misc TCP traffic, for example /usr/bin/wget.anondist-orig (circumvent
```

Interview

"I could take, consider, implement, and share. FOSS was a revelation!"

LINUX FORMAT DVD
THE BEST DISTROS & MORE
EVERYTHING YOU NEED TO GET STARTED WITH LINUX
KODI 14.1
64-bit
The ultimate home media centre distro
MINT 17.1
64-bit
The greatest distro!
FREE eBOOK WORTH £5.99!
With the all-new Cinnamon desktop and long-term support for total peace of mind
LATEST PATCHES & FIXES

On your FREE DVD

Mint 17.1, Kodi 14, Scientific Linux, TinyCore and more!

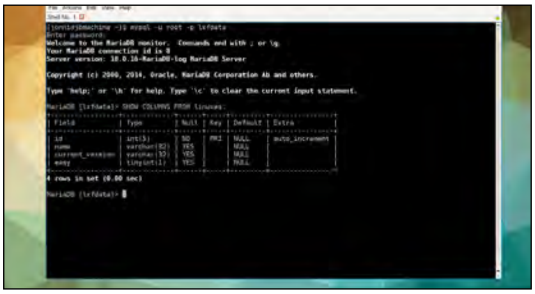
» The best distros money can buy.

PLUS: 164-page Android eBook **p96**

Treat yourself or a loved one to an LXF subscription! **p34**

Don't miss...

Coding Academy



```
[jonni@jbmachine ~]$ mysql -u root -p lxfdata
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 8
Server version: 10.0.16-MariaDB-log MariaDB Server

Copyright (c) 2000, 2014, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [lxfdata]> SHOW COLUMNS FROM linuxes;
+-----------------+-------------+------+-----+---------+----------------+
| Field           | Type        | Null | Key | Default | Extra          |
+-----------------+-------------+------+-----+---------+----------------+
| id              | int(5)      | NO   | PRI | NULL    | auto_increment |
| name            | varchar(82) | YES  |     | NULL    |                |
| current_version | varchar(32) | YES  |     | NULL    |                |
| easy            | tinyint(1)  | YES  |     | NULL    |                |
+-----------------+-------------+------+-----+---------+----------------+
4 rows in set (0.00 sec)

MariaDB [lxfdata]>
```

Regulars at a glance

› **Our subscriptions team is waiting for your call.**

Tutorials

Congratulations. This browser is configured to use Tor.

› **Cry me a river, NSA.**

CANONICAL PHONES HOME

Ubuntu Phone is finally here

The first Ubuntu Phone might not be all that, but it's a promising start.

› The BQ Aquaris Ubuntu Edition is the first of many Ubuntu Phones... we hope.

It's been a long time coming but the world's first Ubuntu Phone has been released. The BQ Aquaris E4.5 Ubuntu Edition has recently gone on sale and it looks like there was plenty of pent up demand for a Canonical-flavoured handset. Although so far it has only been sold in limited 'flash sales', the first such sale completely sold out in under an hour. It bodes pretty well for Canoncial's fledgling mobile operating system, especially considering that the actual phone isn't anything too special. It comes with a 4.5-inch qHD (960x540) display, a MediaTek quad-core Cortex-A7 processor clocked at 1.3GHz and 1GB of RAM. It also features a 5-megapixel front camera and an 8-megapixel rear camera, which aren't bad specifications by a long shot, but not something to get terribly excited about, though perhaps the relatively low price of €169.90 (around £127) helped drum up interest.

> "Launching a new product is a journey, this is the first step."

A number of networks in Europe such as **amena.com**, Giffgaff, and Portugal Telecom will offer SIM bundles with the BQ Aquaris Ubuntu Edition.

What is worthy of interest is Ubuntu Phone itself, a mobile operating system with plenty to offer and set it apart from Apple and Google's offering.

Ubuntu Phone looks set to shake up the way we interact with our phones and the apps we've installed on them. Rather than laying out all the apps separately in grids as with Android and iOS, Ubuntu Phone is bringing in the revolutionary idea of "Scopes". Ubuntu Phone is looking to integrate apps and services via Scopes into an easy to use interface, so users don't have to scroll through screens of icons. For example the *NearBy* Scope will aggregate local services centred around where you are and what you're doing, giving you reviews of local restaurants nearby, as well as the weather in your area. We've been told that we can expect some big-name apps appearing on the new mobile operating system. When asked Cristian Parrino, VP of Mobile at Canonical, what sort of apps and Scopes we can expect in Ubuntu Phone, he replied "Facebook, Twitter, eBay, Time Out, Yelp, as well as community developed apps working on Spotify, Dropbox. Lots. Just about every digital platform, in some cases the top brand, others from valid alternatives"

› Scopes are an innovative new interface feature that Canoncial hopes will make Ubuntu Phone stand out from the crowd.

Almost 1,000 apps from big names around the world that are coming to Ubuntu Phone.

It seems to us that the BQ Aquaris E4.5 Ubuntu Edition isn't really aimed at the mainstream consumer and is instead aimed at developers, with Canonical keen to get an Ubuntu Phone into the hands of app makers as quickly as possible. What will be interesting to more mainstream consumers is the upcoming Meizu handset.

The Chinese smartphone manufacturer signed an agreement with Canonical late last year, and it looks like its upcoming Meizu MX4 , will be Ubunutu-flavoured. This handset is expected to feature far more impressive high-end components: "octa-core" – quad-core Cortex A17, quad-core Cortex A7 – MediaTek MT6595 SoC, 5.36-inch 1920x1152 IPS display, 20MP camera, 4G-LTE and 2GB of RAM. If the specifications are correct, then we should see an Ubuntu Phone that can go toe-to-toe with flagship devices from Apple and Android manufacturers.

When asked Parrino told us that "launching a new product is a journey, this is the first step", and that for the time being Canonical is focusing on early adopters who are bored of iOS and Android, rather than fighting for shelf space among more well known brands. Meizu's Ubuntu Phone could be the start of the real fight.

OPEN SOURCE FUNDING

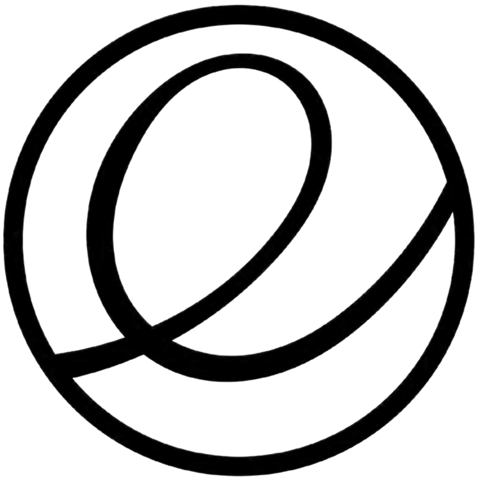
ElementaryOS gets into hot water

Poor choice of words pits elementaryOS against its users.

Wener Koch hasn't been the only one to struggle raising funds. The stylish distro elementaryOS latest version may have been downloaded over 2 million times, but it seems 99.875% of those, download it without paying a single penny.

As with other Linux distros elementaryOS is free to download, but also asks for voluntary donations to help fund the project. Previously when downloading the ISO of the distro, a text box was included where you could type in an amount of money you'd like to donate, with $10 being the default setting. In a bid to get more donations it has now changed the download process, with four buttons ($10, $25, $50 and Custom) which you need to select before you download. If you'd wish to download for free you need to select Custom and then type in $0.

While the additional hurdle to download elementaryOS is minimal, we can at least sympathise with the team's plight of getting funding. However, what didn't help was the rather accusatory blog post (**http://blog.elementary.io/post/110645528530/payments**) that accompanied the news which managed to rile up elementaryOS's fans. A particularly ill judged line stated that "We want users to understand that they're pretty much cheating the system when they choose not to pay for software." It has since been deleted, but the internet's memory is long, and supporters of elementaryOS quite rightly didn't like the accusation.



› Have the folks behind elementaryOS taken their supporters for granted?

OPEN SOURCE FUNDING

GPG gets funded

Fundraisers come to the aid of Werner Koch, the man behind Gnu Privacy Guard that protects us all.

If you value the privacy and protection of your emails and use email encryption then there's a good chance you use *Gnu Privacy Guard*, or software that's based on it. Journalists, companies, dissidents and whistle blowers like Edward Snowden around the world use software, however it was recently revealed that the man behind the code, Wener Koch, was running out of the money.

Although he continues to patch and update *Gnu Privacy Guard* from his home in Erkrath, Germany, in an interview with Pro Publica (**http://bit.ly/16uti0Z**), Koch revealed that he was running out of money. A fundrasing campaing launched in December raised $43,000, quite a bit less than the $137,000 goal. However since the article highlighting Kock's difficulties was published a number of individuals and organisations have rallied to his support, pledging funds for the man behind the software so many of us use.

The Linux Foundation's Core Infrastructure Initiative – that was established for just this eventuality – awarded him a grant of $60,000 and Facebook and Stripe (an online payment processor) have each pledged to donate $50,000 a year to the project. His donations page at **https://gnupg.org/donate** has also been inundated with donations, and if you'd like to thank Koch for his hard work on such an essential tool, please visit and give what you can.

Newsbytes

» Is nothing sacred? If you've got a hard drive made by Western Digital, Seagate, Toshiba or other top manufacturers then be warned: the U.S. NSA has apparently found a way of inserting spying software into those drives. Security firm Kaspersky discovered the software in PCs from over 30 countries, with the most cases being seen in Iran, Russia, Pakistan, Afghanistan, China, Mali, Syria, Yemen and Algeria. **http://blog.kaspersky.com**

» CrunchBang, the lightweight distro that's also known as #!, is no more, with Philip Newborough, announcing in a forum post titled 'The end' that "I have decided to stop developing CrunchBang. This has not been an easy decision to make and I've been putting it off for months. It's hard to let go of something you love." The reason for the decision is apparently because Linux has moved on, and there's no longer a place for CrunchBang. "I honestly believe that it no longer holds any value." **http://crunchbang.org**

» Although Newborough feels there's no longer a place for CrunchBang, others disagree. Almost as soon as the end of CrunchBang was announced, a new project cropped up, known as CrunchBang Plus Plus (or #!++). The aim is to continue the legacy of CrunchBang, keeping it light weight while getting a reskinned user interface and updated Jessie packages. **https://crunchbangplusplus.org**

CrunchBang is dead. Long live CrunchBang Plus Plus!

» Live Patching is coming to the Linux Kernel. Although there have been proprietary live patching features in the past, along with distros such as Red Hat and SUSE bringing their own implementations, the different approaches to live patching were proving problematic. Now a joint approach has been agreed on to create infrastructure for live patching, including an API for kernel modules containing the actual patches, and API/ABI for userspace to be able to operate on the patches.

Comment

LibreOffice 4.4 release

Michael Meeks

Recently we released *LibreOffice 4.4*. By the time you read this it should be ready for wide-spread home usage, and is well worth a try.

For me the highlight is Jan Holesovsky's work (for Collabora) on growing and encouraging the User Experience community. In this release lots of improvements have been made across the suite; such as crisping up the look of the default menu and tool-bars.

This work was based on user statistics indicating which items are more frequently used. But if somehow your favourite icon is not visible by default – it's only a couple of clicks to show it again.

We have a new colour selector, an improved side-bar, better rulers, and we've finally finished making all dialogues dynamically resizable. *Impress* is also much improved, with a number of really odd quirks fixed in the *Impress* master page editing, and a set of pretty new templates to match.

Naturally there are a slew of other new features. Interoperability is a key focus, with lots of work done on improving round-trip OpenXML support (for collaboration with less-enlightened users). Another major win is the inclusion of Google's great Carlito and Caladea, fonts which provide metric compatibility for modern Microsoft documents, much improving layout. We also now interoperatate with OneDrive as well as recently *SharePoint*.

One last good taste is the continued drive to rescue data from proprietary file formats and bring them into ODF: we now have filters for: Adobe *PageMaker* files and Sony e-Books. Do download it, have a play, and get involved.

› Michael is a pseudo-engineer, semi-colon lover, SUSE *LibreOffice* hacker and amateur pundit.

Hitting the mirrors

What's behind the free software sofa?

MAGEIA 5

If all has gone according to plan you should be able to download Mageia 5 by the time this issue goes on sale. Recently the second Beta of Mageia 5 was delayed, which could impact on the final release. The previous major version of the Mandriva fork was very well received, so we have high hopes for this release as well. Find out more at **www.mageia.org**.

› As long as there haven't been any more delays, Mageia 5 should be available to download

CLEAROS 6.6.0

ClearOS 6.6.0 "Community" edition is now available. The latest version of the CentOS-based distribution introduces WPAD, QoS, YouTube School ID support, an upgrade to the Intrusion Detection engine, and ISO-to-USB key support. ClearOS is aimed at cloud servers and gateways for homes, hobbyists and small organisations. A handy web-based interface grants you easy access to the range of features ClearOS provides and you can find out more and download the latest realise at **www.clearfoundation.com**.

› ClearOS brings a number of useful features and security tools for students and organisations.

NETRUNNER 15

Netrunner 15 brings huge changes to the Kubuntu-based distro, and now comes with the KDE Plasma 5.2.0 desktop by default. This version has been redesigned from the ground up, so a fresh install is necessary. KDE Frameworks 5.7 and Qt 5.4 are included, along with a number of new applications and features. Find out more at **www.netrunner.com**.

› V15 is a major update.

UBUNTU 15.04 FINAL BETA

The final beta of Ubuntu 15.04 will be hitting the mirrors for downloads a few days after this issue goes on sale. If you're a confident beta tester or you just can't wait to see what's coming with Ubuntu 15.04 then you can download the beta release from **www.ubuntu.com**. Make sure you back up all your files before you try it out.

› The final beta before the full release is available to download.

COMPETITION

WIN! AN APPLE WATCH

KEEP UP WITH THE TIMES IN TERMS OF TECHNOLOGY AND STYLE WITH THIS CHANCE TO OWN 2015'S MUST-HAVE GADGET…

5 APPLE WATCHES MUST BE WON!

Smart watches are one of the most exciting new technology categories around and there's surely no more eagerly awaited product than the Apple Watch.

Thanks to a combination of Apple's innate ability to make the latest technology accessible and the unrivalled personalisation that wearable technology provides, the Apple Watch is at the top of many people's wish lists.

With an Apple Watch, notifications appear on your wrist, included apps track your physical activity and exercise sessions, the built-in heart rate sensor enables you to monitor your workout performance, and fashionistas can customise the face of the watch to their heart's content. Naturally, the Apple Watch also integrates beautifully with an iPhone and other Apple devices…

All you have to do to be in with a chance of owning one of these remarkable gadgets is answer the following question:

WHAT IS THE NAME OF THE APPLE WATCH'S BRITISH DESIGNER?

A STEVE JOBS

B TIM COOK

C JONATHAN IVE

If you think you know the answer, simply visit **http://bit.ly/watchcompo** to enter. Good luck!

ENTER ONLINE NOW FOR FREE AT: http://bit.ly/watchcompo

The closing date for entries is 1 May 2015. Only residents of the UK and Republic of Ireland can enter this competition. After the closing date, five winners will be drawn at random from the correct entries. Only one entry per household permitted; multiple entries will be disqualified. See **www.futureplc.com/competition-rules** for full terms and conditions.

Mailserver

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

More mind games

Neil Mohr imagines a world without Richard Stallman. Of course, it's interesting to imagine a world without open source, but RMS is nowhere near as important as he would have us believe. Open source was around long before the Free Software Foundation was created. And, being a completely European thing, there's no reason to assume Linux would not have appeared in the absence of a Richard Stallman. Finnish Linux, for instance, is based on Dutch Minix.

And Apple *et al* would be fine – the BSD history has absolutely nothing to do with RMS – indeed, it pre-dates him (as does another major free project: Wikipedia [does it? – Ed?]). RMS started his campaign for Free Software in the late seventies.

In fact, if you want to imagine a world without Linux, you only have to tweak the dates of the BSD lawsuit slightly. If it hadn't started, or had been settled a bit earlier, then Linus would have used BSD and Linux would never have been born.

RMS may be important in the history of open source, but he was a child of his time and there were many others. This cavalier attitude to history is also shown in the article on filesystems, where it says "By the late 80s filesystems that enabled proper directories were required". That may have been true of PCs, but Unix/Linux is a mini-computer operating system and directories were around in the 70s.

Anthony Youngman, via email.

Neil says: I did leave some small print that said "We realise someone else would have championed the philosophy, but as well as katana-wielding Stallman?" I also said "this was just a silly academic exercise" so again, it was in no way serious, but entirely silly. Also I wasn't obsessing about Linux, but the concept of 'open source', so as you point out yourself BSD would have been included in that group and Apple, or to be more historically correct as that's so important, NeXT Step would have struggled to develop its BSD-based OS. So no, OS X and iOS wouldn't have appeared. But no, RMS disappearing from history wouldn't have stopped open source being a force in the world, but that was never my point.

Letter of the month

Helping hand

Years ago you published something that I have found very helpful. Actually lots of things; but this one in particular. I am sure you are not the only people to know about how to gracefully reboot a Linux computer with the GUI frozen, or with some similar problem, but I heard about this useful feature from you. It has gotten me out of a fair few hangs.

Unfortunately, I have a poor memory for somewhat random sequences of characters, so I wrote a reminder through a small HTML file to use as my browser home page. Although creating the file did drill the sequence into my memory, this way it is always there. So, in the spirit of FOSS and Copyleft, I've created a trivial contribution to other people's enjoyment of computers, without the limitations of greed so often found in this world. Enjoy, and thank you.

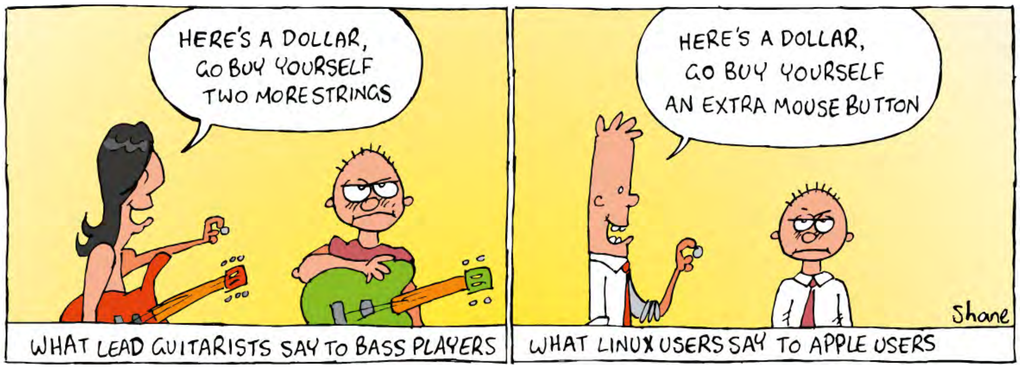
John Paterson, Australia,

Neil says: For those not in the know you can hold Alt and the SysRq key then slowly type REISUB to force a controlled shutdown. This accesses a low-level message system in the kernel, which will only fail if *init* has died or there's been a kernel panic. [No General Panic jokes – Ed.] And the file will reside forever on **www.linuxformat.com/REISUB.html**. Each key stroke does this:

- **R**: Switch the keyboard from raw mode to XLATE mode.
- **E**: Send the SIGTERM signal to all processes except *init*.
- **I**: Send the SIGKILL signal to all processes except *init*.
- **S**: Sync all mounted filesystems.
- **U**: Remount all mounted filesystems in read-only mode.
- **B**: Immediately reboot the system, without unmounting partitions or syncing.

Helpdex

shane_collinge@yahoo.com



HERE'S A DOLLAR, GO BUY YOURSELF TWO MORE STRINGS
WHAT LEAD GUITARISTS SAY TO BASS PLAYERS
HERE'S A DOLLAR, GO BUY YOURSELF AN EXTRA MOUSE BUTTON
Shane
WHAT LINUX USERS SAY TO APPLE USERS

As for the history of filesystems in 250 words, I'll let Jonni cover that responding to the next letter.

Horrid histories

I was surprised to read a really poorly researched and misleading footnote on page 49 [*Filesystems: The Next Generation*, LXF193] on a brief history of file systems. Prior to 1968, there were a number of filesystems in use by various computer manufacturers but the first, that I am aware of, that incorporated most of the features that we would expect in a modern file system was implemented at Bell Labs/MIT in the Multics system. Any history of file systems which omits a mention of Multics can't have been properly researched.

John Hunter, via email.

Jonni says: Unfortunately, there was some context missing in that boxout, which was meant to be taken as an overview of how we got to where we are, rather than an exhaustive account of filesystems throughout the ages. But history is important and you are quite correct – hierarchical filesystems have been around since Multics, and they even included support for long filenames and permissions which the outside world never got to play with until the mid 1990s.

I should have made clear that I was only discussing the situation for home computers, which, while only a fraction of the history of computing, does provide some nice milestones which many readers will remember.

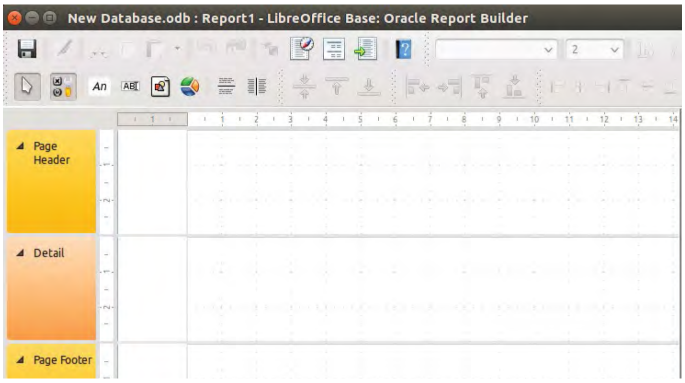
Report MIA

I feel I must gently take you guys to task. I am a long time reader of your excellent journal and a strong supporter of OS and Linux, but I sometimes feel your views of how competitive Linux is against Windows are a little overstated.

As an example I recently set up a new system for a local church, whose ageing Windows XP Toshiba needed to be decommissioned. Part of the job would be the production of data in spreadsheet form, and it seemed to me that a suitable database front-end would be very valuable. I therefore had a quick look at *LibreOffice* and as expected I found that it was; *Base* would connect to data in spreadsheet form.

However, when I tried to produce a report using the Report Wizard it would only produce one rather useless layout, and any attempt at editing it or going direct to the Report Editor failed. A bit of hunting around in the Software Manager to find out if Report Generator was a separate install showed nothing.

Puzzled I did some searching to find a statement from a Ubuntu developer that said: "Report Generator had been excluded because it had too many dependencies." At first I thought this was a restricted situation, but SolydK is Debian based and had the same problem. Annoyingly, Windows had no such issue.


New Database.odb : Report1 - LibreOffice Base: Oracle Report Builder

› Look ma, no dependencies required but your mileage might vary.

› Imaginging a world without RMS: We're not trying to rewrite history, BSD is fantastic too!

I eventually solved the problem by hunting through the long list of dependencies until I found the Report Generator and installed it plus all dependencies. In LXF193 Andrew Procter had a related problem about databases. Your reply was helpful but again there is the implication that *LibreOffice* will do all that is required of a database. This is not the case.

My point is this: anyone like Andrew or any organisation wanting to do the same sort of thing could easily be put off by this decision. So guys how about a tutorial on how to use *LibreOffice* with data?

David Connell, via email.

Neil says: How dare Ubuntu not support every piece of software in the known universe! I jest, it seems this is more an issue with whoever maintains the *LibreOffice* repositories for Ubuntu rather than Ubuntu itself.

The *Report Generator* worked out of the box with Mageia 4 and with Ubuntu 14.04/14.10 we just needed to use:

```
sudo apt-get install libreoffice
libreoffice-report-builder
```

However, we do take your point, that the "out of the box" experience isn't always what it should with open source software sometimes. The truth is though that more often than not most of the mainstream Linux distros do provide a smooth experience and isn't that the most important aspect to highlight?

You gimps

Having scanned hundreds of old photos retrieved from various

»

WHAT'S YOUR "TOP 100" LINUX TOOLS?
GIMP, AUDACITY, KTOON AND BLENDER
"TOP 100"...?
BINARY IS MY TOP ONE MATHEMATICAL PARADIGM
Shane

» family members and done what I thought was a reasonable job of restoring them you then go and include *GIMP: The Complete Guide* on the DVD. Darn it, now I have to go through them all again! I hate you! I hate you! I hate you!

Trevor Dipper, UK

Neil says: It's our pleasure Trevor. We hope the *LibreOffice* eBook last issue doesn't have you redoing all of your documents or the Android bookazine this issue doesn't have you remodelling your Android home screen. Do we dare even mention the Chromebook eBook next issue?

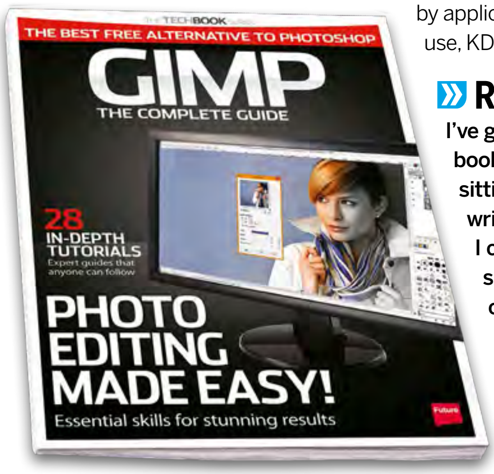
Weighting

I have a question. How can the Cinnamon desktop be the overall winner if it trails behind KDE in overall points [*see the table below*]. While Cinnamon gets a total of 28 points, KDE gets a total of 33 points, yet Cinnamon is still the winner?

Peter Schroeder, via email

Neil says: Nice table! You've got a totally valid point, as maths never lies, which is something our tamed maths PhD man **Jonni**, likes to constantly remind us of, as well as attempting to recite Pi to infinity. The fact is those categories are created especially for every Roundup and they don't necessarily have the same weighting of importance. Cinnamon and KDE score very closely, with KDE clearly having better documentation and application support.

The important thing to remember here is that KDE has been around as long as time itself. So **a)** Its documentation is going to be excellent, more so than Cinnamon's and **b)** It's going to have more applications that support it for a similar reason. The question is: does this make KDE the best choice for everyone? More to the point does *Linux Format* think this makes KDE the best choice? We say no, but if you love documentation and support by applications you may never use, KDE is the choice for you.


The TECHBOOK Series
THE BEST FREE ALTERNATIVE TO PHOTOSHOP
GIMP
THE COMPLETE GUIDE
28 IN-DEPTH TUTORIALS
Expert guides that anyone can follow
PHOTO EDITING MADE EASY!
Essential skills for stunning results
Future

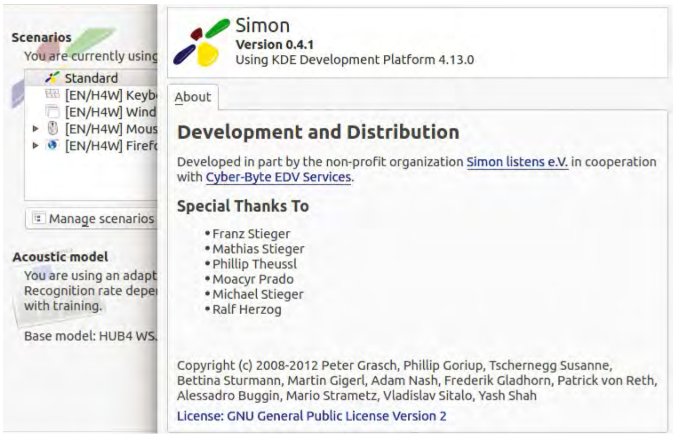
› **If you've enjoyed your *Gimp* bookazine, keep an eye out for more to come.**

Recognise this

I've got an amazing idea for a book, but the thought of sitting at a keyboard to write it fills me with gloom. I could dictate it to a skilful secretary but could my computer do the job instead? The software to translate speech to text has existed for a long while; *Dragon Naturally Speaking* is the standard commercial program. A quick check on Synaptic tells me that there's also a FOSS equivalent that's called *Simon*.

The developers of *Simon* are rather shy about their product, though: there's virtually no documentation only the built-in help. How do you set it up? What kind of microphone does it need? How does it have to be 'trained' to adapt to my kind of spoken English? How accurate is it? How does it handle things like punctuation? What kinds of text formats does it output? I'm sure other *LXF* readers would find this very useful.

Maurice George, Lancashire



You are using an adapt
Recognition rate depe
with training.

Base model: HUB4 WS.

Simon
Version 0.4.1
Using KDE Development Platform 4.13.0

Development and Distribution

Developed in part by the non-profit organization Simon listens e.V. in cooperation with Cyber-Byte EDV Services.

Special Thanks To

- Franz Stieger
- Mathias Stieger
- Phillip Theussl
- Moacyr Prado
- Michael Stieger
- Ralf Herzog

Copyright (c) 2008-2012 Peter Grasch, Phillip Goriup, Tschernegg Susanne, Bettina Sturmann, Martin Gigerl, Adam Nash, Frederik Gladhorn, Patrick von Reth, Alessadro Buggin, Mario Strametz, Vladislav Sitalo, Yash Shah

License: GNU General Public License Version 2

› **For a FOSS alternative to *Dragon Naturally Speaking* try *Simon*.**

Neil says: It turns out we have in fact covered Simon in a Roundup of [*p24*, **LXF188**]. It won out over the other FOSS options and frankly if you want speech recognition it's the only viable option you have. Here's what David Hayward had to say at the time: "The *LXF* voice and speech recognition program of choice boils down to a shouting match between *Speech-App* and *Simon*, as both provide the user with a better overall voice recognition system over the remaining three. In this Roundup, *Simon* comes out as the overall winner with *Speech-App* closely following behind. *Simon* is a more complete and all-round program, and one that will no doubt take over as the clear Linux competitor to the likes of *Dragon Naturally Speaking*."

From that Roundup we found *Simon* was an easy package to install, since it can be found in the Ubuntu *Software Center* and requires nothing else but searching for and clicking the Install button.

Setting up is another thing though, as *Simon* is a little more involved but quite clever to boot. Starting the program initiates a wizard that enables you to configure your voice profile, server profile, test the sound input, output, and volume settings.

When we tested *Simon* with people using strong accents we even found it to be excellent with the odd extra bit of recognition training. So hopefully that should satisfy your recognition needs, let us know how you get on! LXF

Schroeder's Table of Chastisement

Category	Cinnamon	KDE	Gnome	Mate	Enlightenment
Install & Support	5	5	5	4	2
Default Applications	3	5	5	3	1
Appearance	5	5	5	3	1
Help & Documentation	2	5	4	3	2
Performance	4	4	3	5	5
Extensions/Add-ons	4	5	5	3	2
Configurability	5	5	2	4	4
Total Points	**28**	**33**	**27**	**27**	**19**

Write to us

Do you have a burning Linux-related issue you want to discuss? Want to let us know what inventive uses you've been finding for your Raspberry Pi, or suggest future content for the magazine? Write to us at *Linux Format*, Future Publishing, Quay House, The Ambury, Bath BA1 1UA, or email **lxf.letters@futurenet.com**.

United Linux!

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

Find and join a LUG

» **Blackpool Makerspace** Meet every Saturday, 10am-2pm. At PC Recycler, 29 Ripon Road FY1 4DY. **http://blackpool.lug.org.uk**

» **Bristol and Bath LUG** Meet on the 4th Saturday of each month at the Knights Templar (near Temple Meads Station) at 12:30pm-4pm. **www.bristol.lug.org.uk**

» **Edinburgh LUG** Meet on the first Thursday of the month at The Southsider, West Richmond St, Edinburgh. **www.edlug.org.uk**

» **Hull LUG** Meet at 8pm in Hartleys Bar, Newland Ave, 1st Tuesday every month. **http://hulllug.org**

» **Lincoln LUG** Meet on the third Wednesday of the month at 7:00pm, Lincoln Bowl, LN4 1EF. **www.lincoln.lug.org.uk**

» **Liverpool LUG** Meet on the first Wednesday of the month from 7pm onwards at the Liverpool Social Centre on Bold Street, Liverpool. **http://liv.lug.org.uk/wiki**

» **Manchester Hackspace** Open night every Wednesday at their space at 42 Edge St. **http://hacman.org.uk**

» **Surrey & Hampshire Hackspace** Meet every Thursday from 6:30pm at the Games Galaxy in Farnborough. **www.sh-hackspace.org.uk**

» **Tyneside LUG** Meet from 12pm, first Saturday of the month at the Discovery Museum, Newcastle. **www.tyneside.lug.org.uk**

We're jamming!

The Raspberry Jamboree is almost upon us.

Back in 2013 the worldwide phenomenon of Raspberry Jams were still in their infancy, but the irrepressible Alan O'Donohoe (**@teknoteacher**) was moving on to a much bigger event: The Raspberry Jamboree. The Jamboree is an annual event for Raspberry Pi fans from across the world, people travel from as far as Dubai to take part and show their projects and talk about how the Raspberry Pi has become an integral part of their workflow.

The first Jamboree was on the outskirts of the Education Innovation Conference and Exhibition (EICE) at Manchester Central, and merely a small conference room full of stalls. Speakers and delegates mingled in an informal manner and attendance was around 200 people (which was more than those attending the EICE conference).

The following year, Alan negotiated to integrate the Jamboree into EICE and we saw stalls dotted around the exhibition. The Raspberry Pi Foundation ran its hands-on projects with the OCR examining board, who has supported the Foundation's mission to enable children to learn computing. The 2014 Jamboree also introduced panel discussions with notable members of the community, such as Ryan Walmsley (RyanTek Robotics), Andrew Mulholland (Pi-LTSP), Jason Barnett (Cyntech).

For 2015, the Jamboree moves to Our Lady's Catholic High School, Preston where Alan teaches. It will be a three-day event, from Friday March 20 to 22. Friday will be a teacher-only CPD day, but there's a Family Hack in the evening. The Jamboree itself will run on Saturday and Sunday. Tickets are currently available via an Eventbrite page **http://bit.ly/AoDEvents**. LXF

› **The Raspberry Pi Foundation are firm supporters of the Raspberry Jamboree, so join the fun.**

Community events news

Edinburgh Mini Maker Faire
The capital of Scotland will play host to its own Mini Maker Faire on April 19 at Summerhall. On show will be projects that cover the broad spectrum of maker culture, with electronics rubbing shoulders with wood work and tailoring. This is a great event for seeing some immense talent using technologies such as the Raspberry Pi and Arduino.
For more information head over to the official website.
http://makerfaireedinburgh.com

Preston Raspberry Jam
Preston's monthly Raspberry Jam is growing, says Martin Bateman a host for the event. It now includes regular presentations from Sonic Pi competition finalist, Joshua Lowe, who has been hacking and writing music with Raspberry Pi and Sonic Pi 2. Preston Raspberry Jam is a mix of presentations, workshops and social event, and takes place on the first Monday of each month.
www.raspberrypi.org/jam

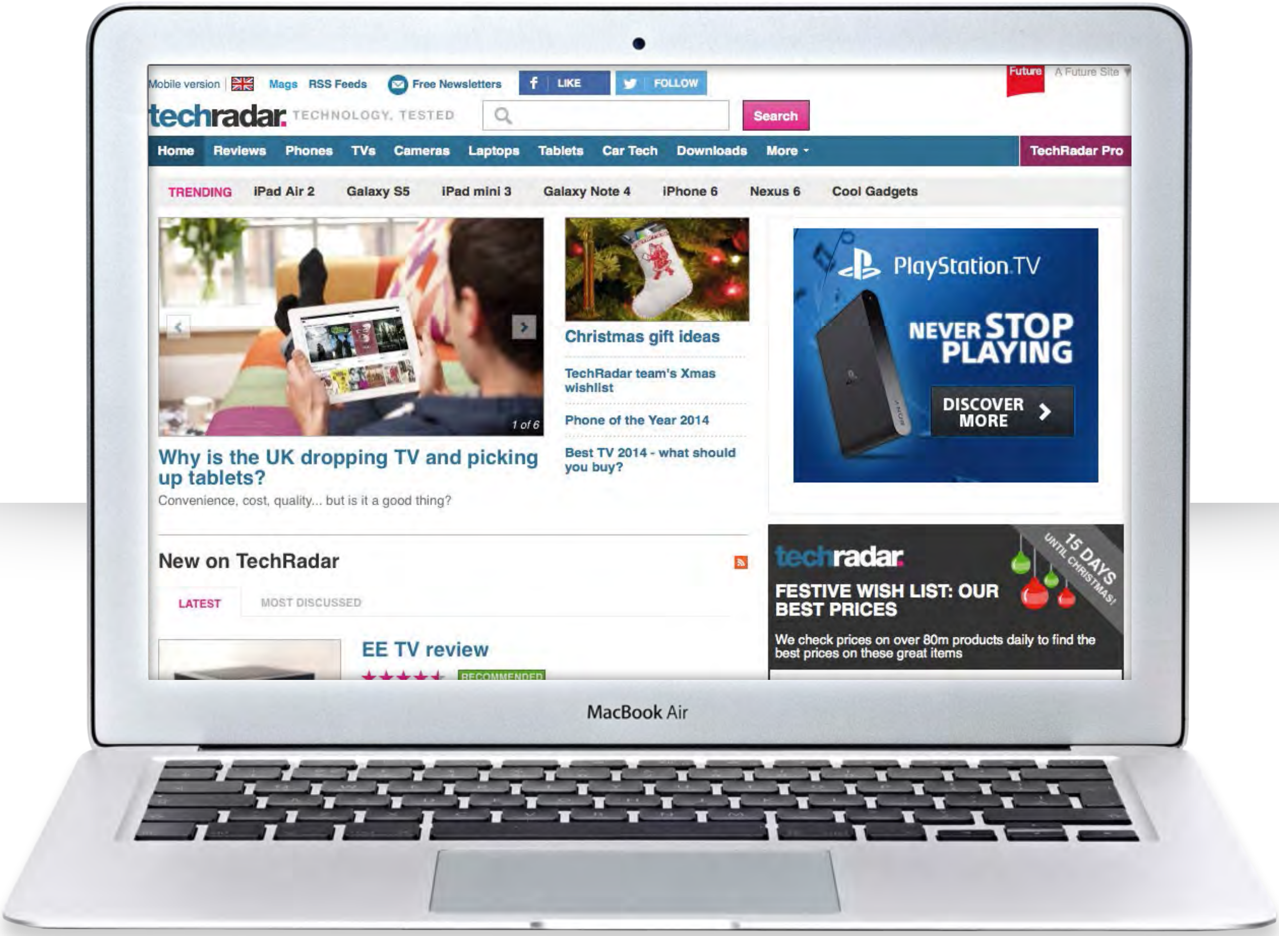
NoSQL Matters
In the last two years noSQL databases have come to the fore of web technologies and not a week goes by where we don't hear about something new from the likes of *MongoDB* and *Riak*.

In that light, NoSQL Matters, Dublin takes place June 3-4 with day one hosting a developer training day and day two featuring a full conference track of workshops and talks on the latest NoSQL advancements.
If you are a database or web developer looking for continued development training this event is a must. And, of course, there's the chance for a Guinness or two at the various social events. See the website for details.
http://bit.ly/NoSQLMatters2015

Future

techradar.

TECHNOLOGY. TESTED.



Why is the UK dropping TV and picking up tablets?

Convenience, cost, quality... but is it a good thing?

Christmas gift ideas

TechRadar team's Xmas wishlist

Phone of the Year 2014

Best TV 2014 - what should you buy?

New on TechRadar

EE TV review

techradar

FESTIVE WISH LIST: OUR BEST PRICES

We check prices on over 80m products daily to find the best prices on these great items

15 DAYS UNTIL CHRISTMAS!

VISIT TECHRADAR, THE UK'S LEADING TECH NEWS & REVIEWS WEBSITE

- Up-to-the-minute technology news
- In-depth reviews of the latest gadgets
- Intensive how-to guides for your kit

www.techradar.com

twitter.com/techradar facebook.com/techradar

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

HP ZBook 15u G2

A lightweight, no-frills budget mobile workstation that HP ships with Linux as a native option and **Juan Martinez** thinks is worth a look.

Specs

- **OS:** Ubuntu
- **CPU:** Intel Core i5 5200U (2.2GHz)
- **GPU:** AMD FirePro M4170 1GB, Intel HD Graphics 5500
- **Display:** 15.6-inch LED 1,920x1,080
- **RAM:** 4GB DDR3
- **HDD:** 500GB
- **Comms:** Intel 7265ac + Bluetooth 4.0LE
- **Ports:** 3x USB 3.0, DisplayPort, VGA, 720p webcam
- **Size:** 375.5x253.6 x21.42mm
- **Weight:** 1.91kg

Unlike the majority of consumer laptops on the market, mobile workstations are built to provide desktop performance on portable frames. This means organisations are likely to sacrifice thickness and heft in exchange for storage capacity, processing speeds and graphics power. Most workstations offer no-frills design and don't break in more avant garde consumer-friendly features (for example: you won't see them flip over and turn into tablets). And because these machines are built to withstand the rigours of long workdays, they're likely out of a typical consumer's budget.

Among the best mobile workstations on the market is the boringly-designed Lenovo W540 [*see Reviews, p17* **LXF192**], which weighs 2.52kg, but offers an incredible 15.6-inch, 2880x1620 (3K) resolution IPS display and a high-end 2.7GHz Intel Core i7-4800MQ processor with a Nvidia Quadro K2100M graphics card. Not to be outdone, HP offers the ZBook 17, an expensive workhorse that weighs 3.17kg, is 1.3 inches thick, but can outperform some of the best desktops on the planet. Built with an Intel Core i7 processor, up to 32GB of DDR3 memory and Nvidia Quadro graphics.

Weighing in at just 1.91kg, the 15.6-inch ZBook 15u is one of the lightest workstations on the market. Built with an FHD screen and no-spindle drive, the ZBook 15u is meant to provide business users with the ability to get up and go without sacrificing all of the perks of a desktop workstation. It measures 0.84x14.78x9.98 inches tall – not exactly something you'll want to hold in one hand, but it won't dislocate your shoulder when you sling your laptop bag on after a long meeting.

› **The ZBook 15u provides a smidge more portability and accessibility, than the more powerful Zbook 17.**

Solid worker

The default OS is Windows 8.1 but dig into the component options and HP offers 'Ubuntu Linux' and even a big lump of cash back for the non-Redmond option. The ZBook 15u comes with up to 1.25TB of storage and 16GB of memory with an Intel i5 processor on all standard models. It's ISV certified with AMD FirePro M4170 graphics. You'll never lack for inputs, as the 15u features four USB 3.0 ports, a DisplayPort 1.2, and a media card slot.

There's not much to complain about with the 15u. You realise that you're sacrificing something for this unit when you spend over $1,000 less than you would with competitor models.

For example: the graphics won't be super high. The AMD FirePro M4170 was a mid-range graphics card mid-2014, and it will likely move further down the list as the machine ages. Additionally, the Intel i5 is solid, but most workstations come equipped with i7 processors. You can upgrade, but it will cost extra. Although the machine is pretty to the touch and to the eye, you're not getting anything new here. You won't impress anyone with the traditional keyboard or touchpad, and the screen isn't touch-capable.

If you need a workstation but you don't want to break the bank, the ZBook 15u is an excellent compromise. It offers solid processing speeds, decent graphics, but this is middle-of-the-pack at best. What you're getting is a lightweight workstation with a pretty design that your Chief Financial Officer will happily approve. LXF

LINUX FORMAT Verdict

HP ZBook 15u G2

Developer: HP
Web: http://store.hp.com
Price: From £1,267

Features	9/10
Performance	7/10
Ease of use	9/10
Value for money	9/10

» *Offers the lightest build, lowest cost with a balanced processor and AMD GPU alongside native Linux support.*

Rating 8/10

Dell Precision T5810

A machine that offers more than **Alan Stevens** was expecting and in an attractive form, too.

› **There's plenty of room inside the T5810 for access and expansion.**

In brief...

» There's a fine line between high-end desktop PCs and entry-level workstations. However, with its new Haswell-based processors allied to a choice of highly capable graphics, the Precision T5810 is very much in the latter group.

Specs

- » **OS:** Ubuntu 12.04 LTS
- » **CPU:** Intel Xeon E5-1620 quad-core 3.5GHz, 10M cache
- » **RAM:** 16GB 2133MHz DDR4 ECC
- » **GPU:** AMD FirePro W5100 4GB
- » **HDD:** 1TB SATA
- » **Warranty:** 3-year next-day on-site

Following the release of Intel's Haswell-based Xeons, Dell has acted quickly to adopt the new processor and use it to put extra spring into the step of its popular Precision workstation family. Dual and single-socket workstations will both benefit from the makeover and for this review we decided to check out the Precision T5810, a single-socket workstation that sits at the more affordable end of the spectrum.

Following the standard workstation pattern, the T5810 is housed in a substantial yet compact black and silver tower which, thanks to a couple of design flourishes, manages to somehow look pretty stylish. Power comes from an externally removable 685W AC supply which has two fans built in for cooling with, on the review system, another built onto the video card. The end result was far from silent, but we'd still put it down at the quiet end of the scale for this type of system.

Designed to be customised, access to the T5810 is provided via a lift-off side panel with plenty of room to get to all the important bits and pieces. And that includes an all-new motherboard sporting a single socket for the Haswell-based Intel Xeon processor plus, of course, the latest C612 chipset required to manage it.

Processor-wise, the workstation we looked at was fitted with a Xeon E5-1620 v3 (it's the 'v3' that signifies the Haswell architecture) clocked at 3.5GHz and delivering four cores/eight threads plus a decent 10MB of cache – a good starting point for buyers on a budget. Moreover, those with extra cash to splash can opt for a number of alternatives both from the single-socket E5-1600 v3 family and the E5-2600 v3 line-up, more usually found in dual-socket configurations.

As far as memory is concerned support for the latest DDR4 technology is part of the Haswell update, with eight DIMM slots available on the T5810 motherboard to take up to 128GB, clocked at up to 2133MHz. Ours had a more modest 16GB of ECC-protected memory on-board which is, again, a reasonable starting point without costing the Earth. The DIMMs provided by Dell also left half the slots empty making later additions easy to accommodate.

Budget workstation

Haswell processors aside, it's the video controller that really sets the performance bar in the workstation world with plenty of options here to help make the T5810 fly. Customers with an existing adaptor can save money by leaving this out altogether, otherwise it's a choice between AMD FirePro and Nvidia Quadro GPUs, with a number of models on offer to suit both budget and application requirements. There's room and power for a pair of graphics cards if needed, but the review system had just one in the form of an AMD FirePro W5100. This 3D capable card takes up a single PCIe slot and has four DisplayPort connectors offering support for monitors with up to 4K resolution. In addition there's scope for further improvement by opting for a higher-spec GPU.

Storage is, again, down to the customer to specify. The review system had just a single 1TB SATA disk cabled to the onboard controller, but there's plenty of room to add more and RAID is an option. You can also opt for faster SSD drives both for use as conventional storage and boot purposes, or to act as a performance-boosting cache using Intel CAS-W technology, now available as an option on all Precision workstations. A slimline DVD writer was also fitted on the workstation we looked at and with 10 USB ports (four USB 3.0) plus two on the motherboard there's very little else you can't plug in. An integrated Gigabit network interface also comes as standard and there are five PCIe expansion slots. LXF

Features at a glance

Xeon processor

The new Haswell-based Xeon processors plus DDR4 RAM provide a massive step up.

AMD FirePro

Sports 768 processing cores and 4GB of video RAM which amounts to a mid-range workstation.

LINUX FORMAT Verdict

Dell Precision T5810

Developer: Dell
Web: www.dell.com/uk/business
Price: From £1,573

Features	9/10
Performance	8/10
Ease of use	8/10
Value for money	9/10

» *A very usable system that delivers the performance needed by design and engineering pros on a budget.*

Rating 9/10

Acer Chromebook 13

Say hello to the longest-lasting Chromebook yet, **David Eitelbach** takes it for a lengthy test drive.

› The Tegra K1 processor greatly benefits the battery life of the Acer.

Specs

- **CPU:** 2.1GHz Nvidia Tegra K1 CD570M-A1 (quad-core)
- **GPU:** Nvidia Keplar
- **RAM:** 2GB DDR3
- **Screen:** 13.3-inch, 1,920x1,080
- **HDD:** 16GB SSD
- **Ports:** 2x USB 3.0, HDMI, SD card slot, headphone and mic. jack
- **Comms:** 802.11ac Wi-Fi, Bluetooth 4.0
- **Camera:** 720p HD webcam
- **Battery:** 3,220mAh
- **Size:** 327 x 227 x 18mm, 1.5kg

Compared to most laptops, Chromebooks are a one-trick pony: They're good for web-based productivity and not much else. Acer has refined this simple formula with Chromebook 13, which offers a 768p screen, an outstanding keyboard and a slim and light design. It's also the first Chromebook to feature Nvidia's energy-efficient Tegra K1 quad-core processor, which provides much longer battery life than competing notebooks. Moreover, at £219 the Chromebook 13 is the same price as the Samsung Chromebook 2 [Reviews, p19, **LXF193**], which uses a 1,366x768 display and a dual-core Intel Celeron N2840 CPU.

Sporting a slim, matte white plastic chassis that's refreshingly minimalist. Despite its thin profile, the notebook feels surprisingly sturdy in my hand. On the other side of the lid, the laptop features a 13.3-inch display with a matte, glare-resistant finish. The black island-style keyboard stands in sharp relief against the bone-white deck, and the palm rest offers plenty of room for the wrists. A blue LED at the top right corner glows when the laptop is on.

For the Chromebook 13, Acer ditched Intel's Celeron line of processors for a quad-core 2.1GHz Nvidia Tegra K1 CPU. In our experience, the laptop handled browsing the web, composing a document or streaming music with aplomb, but performance slowed down noticeably when we opened more than a dozen tabs. The Acer also ships with a 16GB solid-state drive, although almost nothing in Chrome OS needs to be saved locally.

At £219, this laptop is on par with typical Chromebooks, which is a surprise given its Tegra K1 processor. By comparison, the Toshiba Chromebook 2 features a dual-core Intel Celeron N2840 processor, 4GB of RAM and a 1080p panel. The Samsung Chromebook 2 11, uses a dual-core N2840 processor, 2GB of RAM and a 1,366x768 display. Annoyingly the US configuration sports a 1080p screen. Instead, European users are limited to a 1,366x768 display on the base model.

Tegra effect

The Chromebook 13 easily handles simple tasks like browsing the web with a few tabs open or streaming 1080p video, but it struggles when streaming music with more than a dozen tabs open. We noticed a half-second delay when typing in the address bar, for example, and Chrome occasionally froze for several seconds whenever we opened a website in a new tab.

Doubling the memory would have significantly improved the Chromebook 13's performance at very little cost. The 11-inch Samsung Chromebook 2, which also uses 2GB of RAM, suffered from noticeable stuttering but the the Toshiba Chromebook 2 with 4GB of RAM, had no trouble streaming music. Thankfully, the Acer's speakers hit almost all the right notes. Audio sounds remarkably clear and well-balanced, and we didn't notice any distortion when the volume was cranked up to the maximum. The laptop's island-style keyboard is one of the better ones I've used on a Chromebook. The chiclet-shaped keys offer plenty of vertical travel and tactile feedback, and the spacing is generous enough that we only made a few adjacent-key errors. We also didn't notice any flex in the keyboard. The spacious touchpad is a joy to use as well. Gestures such as two-finger scrolling and three-finger swiping perform reliably, and the pad doesn't require excessive force to depress. As with most Chromebooks, you have to tap with two fingers to open the right-click menu.

Nvidia promised unprecedented efficiency with the Tegra K1 processor, and it wasn't kidding. The Chromebook 13 lasts longer on a charge than any Chromebook we've seen. While writing the review with the brightness at 50%, watching YouTube videos and streaming music, the notebook lasted a jaw-dropping 8 hours and 56 minutes. That's an hour longer than the Samsung Chromebook 2. LXF

Features at a glance

Nvidia Tegra K1

The latest Arm-based processor from Nvidia helps extend the run time of the Chromebook.

Cool keyboard

Acer has done an excellent job with the keyboard, and typing is easy and swift, and nicely designed.

LINUX FORMAT Verdict

Acer Chromebook 13

Developer: Acer
Web: www.acer.co.uk
Price: £219

Features	9/10
Performance	7/10
Ease of use	9/10
Value for money	9/10

Excellent keyboard and great battery in a slim and light package, but lacks multitasking performance.

Rating 8/10

AMD FX-8320E

A budget octo-core processor that's not from Intel, but is it cheap enough? **Dave James** takes out his wallet and counts the moths.

Specs

» **Socket type:** AMD AM3+

» **Core tech:** AMD Piledriver

» **Clock:** 3.2GHz

» **Turbo:** 4GHz

» **Cores:** 4

» **Threads:** 8

» **Cache:** 4x 2MB L2, 8MB L3

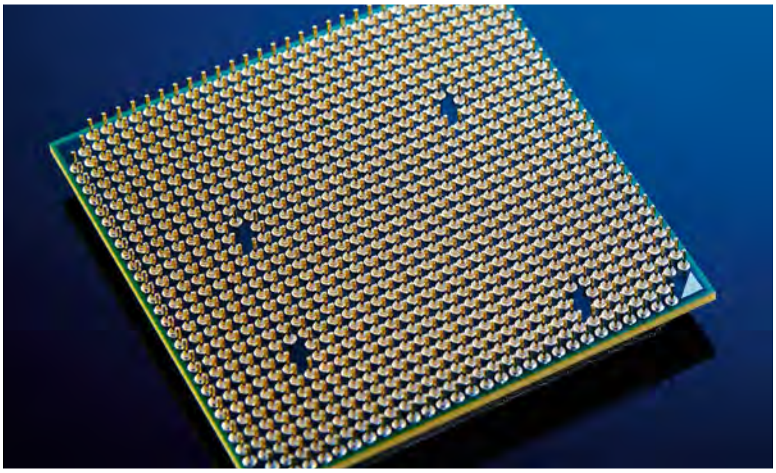
» **TDP:** 95W

» **Process:** 32nm

When AMD tells us it's sending over a new FX-series CPU, we can't help the sudden rush of excitement. It's an automatic response, born of a time when a new AMD CPU had the potential to offer something genuinely competitive. But those days seem long gone. All we get now are half-hearted revisions of increasingly elderly chips.

The FX-8320E is the perfect example of that. AMD released this chip late 2014, along with the FX-8370E as a pair of lower-powered octo-core CPUs for the more power-conscious consumer. These two chips use AMD's Bulldozer processor tech and squeeze it into a 95W TDP. They're able to do this by utilising a lower base clock, retaining the same Turbo clock as their non-E brethren. To that end, this FX-8320E is running at 3.2GHz as standard, with the ability to hit 4GHz as needed. The standard FX-8320's clockspeed sits some 300MHz higher at 3.5GHz.

So far, so good. For 30W less power you only sacrifice 300MHz of CPU horsepower, which seems like a pretty good trade-off. But the fact these CPUs are still running with an outdated version of the Bulldozer architecture makes them seem more like an afterthought than a proper processor release. The Kaveri APU, launched mid-2014, AMD used the latest revision of Bulldozer, codenamed Steamroller. Its next APU, Carrizo, will use the final Bulldozer revision which is codenamed Excavator. This 95W processor then, using the old Piledriver architecture, is two generations behind AMD's top x86 CPU cores. And it doesn't look like it has any interest at all in shifting the FX range over to the Excavator design, despite the IPC (Instructions Per Clock) boosts that both the Steamroller and Excavator tech have over Piledriver.



› **Another 95W Vishera FX chip racing towards obsolescence.**

The cost of saving

What's the game with the FX-8320E then? You'd assume that with the focus on hitting a lower TDP, this chip would be looking at small form factor PCs, but that 95W TDP is still higher than the 84W Haswell Core i5 processors, even the K-series versions. But then there's the price. At just over £100, this is the cheapest eight-core CPU around – even if you baulk at referring to its quad-module design as a full octo-core setup it still sits as the cheapest, eight-threaded CPU you can buy.

In this context, suddenly the FX-8320E looks like a more intriguing purchase. That's especially true if you're sitting on a lower core-count AM3+ chip and feel the need for an upgrade. In performance terms – in both straight CPU and gaming tests – the FX-8320E is evidently behind both the FX-8350 and Intel Core i5-4570, but it's a good £30-50 cheaper than those more powerful chips. And because it's an AMD chip, without the needless limitations imposed on it by overzealous marketing execs (looking at you, Mr Intel K-series), you can get happy with the overclocking. Well, should your chip and board be capable of it anyways.

For the AMD upgrader then, it's not a bad budget option. If you're looking to build an all-new machine though we'd still struggle to recommend an AMD setup. Even though you're getting eight threads of processing power, a resolutely quad-core, unoverclockable Core i5-4570 will still deliver better CPU performance, and in a smaller power envelope too. The Intel platform is also going to be more up to date and not much more expensive either. While AMD's AM3+ chipsets were queuing up for their pensions, Intel's motherboard chipsets were busy fitting themselves out with native USB 3.0 and PCIe 3.0 support, which is handy... LXF

LINUX FORMAT Verdict

AMD FX-8320E

Developer: AMD
Web: www.amd.com
Price: £105

Features	7/10
Performance	6/10
Ease of use	9/10
Value for money	8/10

» *A decent upgrade for an AMD setup, but it's old tech; power-hungry with no native USB 3.0 or PCIe 3.0 support.*

Rating 7/10

Features at a glance

Overclocking

Our sample wasn't happy running above 4GHz. Voltage tweaks helped, but not enough to keep it stable on the 970 board.

Gaming issues

The difference between the Core i5-4570 and this FX chip is nearly 20fps on average at 1080p settings with the same GPU.

Tiny Core 6.0

Shashank Sharma tests a major release of the minimalist distro.

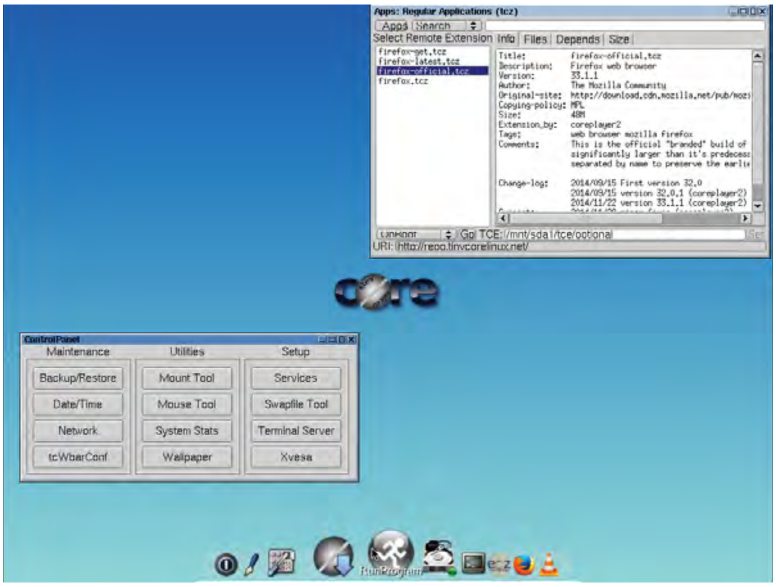
In brief...

» A barebones distro designed for older computers and power users. The small size of the distro means that users are free to mold the distro any way they see fit. While the project ships several editions of varying sizes, users can take the all-purpose Core release and stack apps on top to suit them. See also: SliTaz.

Tiny Core Linux is designed for people willing to put in the requisite effort to build a customised streamlined cholesterol-free distro that suits their needs. The distro has a major new release in which the developers have updated almost every core component, all 15MBs worth! Tiny Core is the miniculest of distros that boots into a graphical desktop and it isn't derived from any of the mainstream distros.

Traditionally only available as a 32-bit ISO, the latest edition also has 64-bit variants and is available in multiple flavours. Besides the 15MB Tiny Core release, there's an even smaller 10MB Core release that ships without a desktop and a 76MB CorePlus variant that includes multiple desktops. The latest release also features a version of the distro for the Raspberry Pi called PiCore, available in multiple editions with images that have a GUI and another with SSH pre-installed for remote administration.

Due to its bare-bones nature, the new release isn't visibly different from previous releases. However, there are substantial differences in the lower support infrastructure. Tiny Core 6 boots off a new kernel and includes a newer glibc package which is used by various apps. The *tce-load* utility has also been tweaked, which can be considered the distro's package manager, to handle the dependencies of large apps like *LibreOffice*.



```
firefox-get.tcz
firefox-latest.tcz
firefox-official.tcz
firefox.tcz

Title:          firefox-official.tcz
Description:    Firefox web browser
Version:        33.1.1
Author:         The Mozilla Community
Original-site:  http://download.cdn.mozilla.net/pub/mozi
Copying-policy: MPL
Size:           48M
Extension_by:   coreplayer2
Tags:           web browser mozilla firefox
Comments:       This is the official "branded" build of
                significantly larger than it's predeces
                separated by name to preserve the earli

Change-log:     2014/09/15 First version 32.0
                2014/09/15 version 32.0.1 (coreplayer2)
                2014/11/22 version 33.1.1 (coreplayer2)

TCE: /mnt/sda1/tce/optional
URI: http://repo.tinycorelinux.net/
```

› **Don't expose this to a real drive without first reading its extensive user guide.**

The distro is quick off the blocks and drops you to a plain desktop running the FLWM window manager. The icon-based application launcher at the bottom of the desktop contains icons to access settings, add packages to the system, mount disks, launch the terminal, and such. When you install apps to the distro, their icons are automatically placed on the launcher.

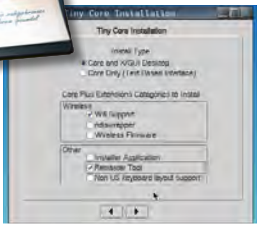
Big on apps

Tiny Core uses its own package format and its repository is flush with hundreds of popular apps. Some apps have recent versions such as *LibreOffice 4.4*, while others such as *Firefox*, *Chromium* and *Thunderbird* have older versions. However *Firefox* has an upgrade script that'll detect the latest upstream version and update accordingly. It worked flawlessly and bumped us to *Firefox 35.0.1*. Adding individual apps is painless, but the distros small footprint means that most apps will download all dependencies. Unfortunately, not all listed apps are installable, such as *VLC*. The version of *VLC* in the repos requires the *libiconv* library whose functionality is now provided by *glibc*.

Despite the availability of apps and conveniences like automated app installers, it still takes some doing to transform Tiny Core into a regular desktop. The distro is designed for users who can't spare the resources required by a regular desktop distro. The recommended Tiny Core edition doesn't even ship with an installer, though it offers many options for making data survive reboots. The distro does have an official installer available in the repos and also pre-installed in the CorePlus edition. This has the useful option to install the distro on a USB disk that you can boot from on older PCs that don't support USB booting.

Getting familiar with Tiny Core requires time and effort. But once you have the system up and running, it's easy to maintain. For example, if you've anchored the distro using the frugal installation method, you can upgrade from the previous release to the latest one by just replacing two files. LXF

Features at a glance



Install Type
● Core and X/GUI Desktop
○ Core Only (Text Based Interface)

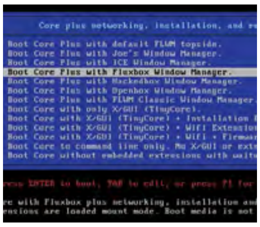
Core Plus Extension's Categories to Install

Wireless
☑ Wifi Support
☐ ndiswrapper
☐ Wireless Firmware

Other
☐ Installer Application
☑ Remaster Tool
☐ Non US Keyboard layout Support

Updated base

The latest distro features kernel 3.16.6 and components such as glibc, gcc and e2fsprogs.



```
      Core plus networking, installation, and re
Boot Core Plus with default FLWM topside.
Boot Core Plus with Joe's Window Manager.
Boot Core Plus with ICE Window Manager.
Boot Core Plus with Flexbox Window Manager.
Boot Core Plus with Hackedbox Window Manager.
Boot Core Plus with Openbox Window Manager.
Boot Core Plus with FLWM Classic Window Manager.
Boot Core with only X/GUI (TinyCore).
Boot Core with X/GUI (TinyCore) + Installation E
Boot Core with X/GUI (TinyCore) + Wifi Extensio
Boot Core with X/GUI (TinyCore) + Wifi + Firmwar
Boot Core to command line only. No X/GUI or ext
Boot Core without embedded extensions with waitu

ress ENTER to boot, TAB to edit, or press F1 for

re with Fluxbox plus networking, installation and
ensions are loaded mount mode. Boot media is not
```

Multiple flavours

The 72MB Core Plus release features more useful tools like wireless drivers and a Pi edition.

LINUX FORMAT Verdict

Tiny Core 6.0

Developer: Robert Shingledecker
Web: www.tinycorelinux.net
Licence: GPL v2

Features	7
Performance	9
Ease of use	6
Documentation	9

» *A must-have upgrade for existing users and a viable option for users holding on to dated hardware.*

Rating 8/10

Netrunner 14.1

An updated release of a distro that makes KDE enjoyable, even for a Cinnamon user like **Shashank Sharma**.

In brief...

» A KDE-based distro designed for desktop users. See also: Kubuntu, OpenSUSE and Mageia.

Although there's no dearth of distros that ship with KDE as the default desktop, Kubuntu seems to hog all the limelight. While we don't have anything against the distro, it doesn't really showcase the true potential of KDE. If you want to taste the best of KDE take a look at the Netrunner distro. It's sponsored by Blue Systems, the same German company that now supports the development of Kubuntu and a handful of KDE projects. The distro is available in two flavours and has recently released an update of their Long-Term Support release.

Netrunner 14.1 is a polished version of the previous release based on Kubuntu 14.04 LTS. It features an updated KDE SC 4.14 desktop, which was the final release in the KDE4 branch. This version of KDE uses an improved version of the new desktop file indexing and search tool *Baloo*. While this release still uses Ubuntu's *Ubiquity* installer, the project's rolling release variant uses the new distro-independent *Calamares* installer.

Apps galore

One of the highlights of Netrunner 14.1 is that it's chock full of apps. All the popular ones, such as *LibreOffice*, *VLC*, *Gimp*, and others are complemented by KDE gems such as *Clementine*, *Krita*, *Karbon*, and *Kdenlive* and many more.

Furthermore, the developers have taken extra care to ensure that the non-KDE apps also adhere to the KDE Plasma theme and don't appear out of place. The result is a stunning desktop that appears coherent despite the multitude of apps.

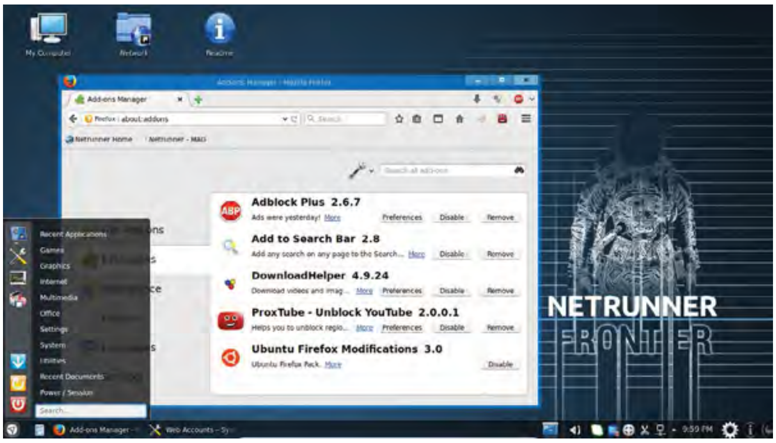
Features at a glance

Updated apps

The LTS distro uses an updated release of the KDE 4 branch and includes new versions of various apps.

Coherent desktop

The distro has taken steps to ensure its variety of apps are well integrated into the Plasma desktop.



› Netrunner lets you use Windows DreamScene animated wallpapers.

Blue Systems also funds the development of several integral components such as the *samba-mounter*, which is a graphical app to easily mount and manage Samba shares. Patched versions of the latest releases of both *Firefox* and *Thunderbird* that blend with the Plasma desktop are another highlight.

If you run Netrunner 14.1 on a desktop with more than 2GB of RAM, the distro will preload *Firefox* on boot. This doesn't seem to have an adverse effect on the distro's boot times and it was quick off the blocks on our 4GB test machine, while loading *Firefox* almost instantaneously.

The browser is equipped with several plugins and extensions including the *Flash* plugin, *Google Talk* video plugin, the *AdBlock Plu*s extension, the *DownloadHelper* extension and more.

While the distros default selection of apps should suffice for most users, you can use the *Muon Software Center* or the classic *Synaptic package manager* to fetch others.

Another inclusion that makes Netrunner a top choice is their homebrew Runners-ID cloud storage service. The distro includes a *Social Accounts configurator* in the KDE System Settings that helps you sign into Runners-ID as well as your accounts on OwnCloud, Facebook and Google. Once configured, these accounts are flawlessly synced with the associated installed apps such as the *Kontact PIM* and the *Telepathy IM* client.

If you've ever been intimidated or even overwhelmed by the endless list of configurable elements in KDE, you'll appreciate the fact that the Netrunner developers have cut down and customised the KDE System Settings list to make it more approachable.

It doesn't take long for one to realise that Netrunner is unlike other KDE distros. Its developers have paid special attention to highlight all the good aspects of the KDE desktop and it shows. The desktop looks beautiful and consistent despite its varied and extensive collection of apps that'll cater to all kinds of desktop users. LXF

LINUX FORMAT Verdict

Netrunner 14.1

Developer: Blue Systems
Web: www.netrunner-os.com
Licence: Various

Features	8/10
Performance	8/10
Ease of use	8/10
Documentation	7/10

» *A wonderfully nurtured distro that deserves its place at the top of the KDE-distros food-chain.*

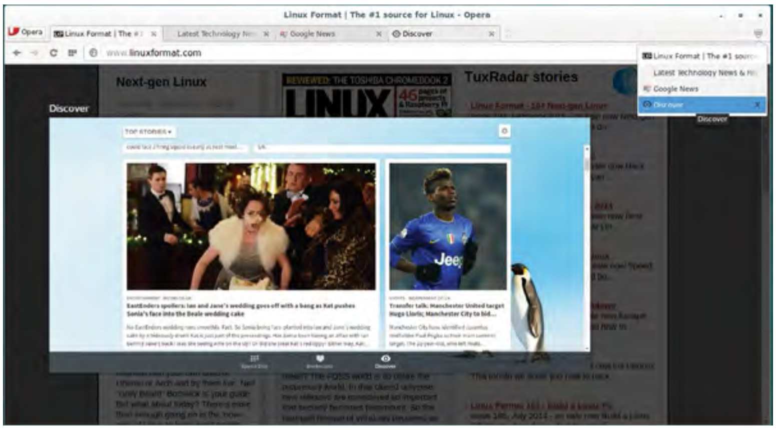
Rating 8/10

Opera 27

More than content with the open source web browsers on offer, **Shashank Sharma** rolls his eyes at the proprietary alternative.

In brief...

» The latest update of the feature-rich but proprietary web browser. The release sadly offers no concrete reason for *Firefox* or *Chromium* users to switch allegiance. It's a competent browser but still has a lot of catching up to do to be in the same league as its more famous open source alternatives. See also: *Firefox* and *Chromium*.



› Apart from the odd feature, Opera can barely hold a candle to Firefox.

The release of *Opera 26* in December 2014 marked the return of major releases of the browser to Linux. The latest release, *Opera 27* continues that trend and was released for Linux alongside versions for Windows and Mac OS X. Officially, the binaries for *Opera 27* are available only as .DEB and that too is just for 64-bit installations, but Opera's team-lead for the Linux initiative maintains a personal repository with RPMs for Fedora and OpenSUSE.

Opera 27 features a handful of new interface and backend changes since the previous release. It's powered by the latest *Chromium/Blink* layout engine. The browser has also adopted *Chromium*'s *Pepper Flash* player along with the associated *Pepper* Plugin API. This is in tune with *Chromium* dropping the use of all NPAPI plugins like the *Adobe Flash Player*.

The Bookmarks Manager has also been tweaked and the developers have fixed issues with context menus and empty bookmark folders. *Opera*'s Bookmarks Manager is more visual compared to other browsers and includes the usual abilities to sort, search, filter and manage bookmarked links. An interesting feature is the browser's ability to share selected bookmarks or even entire bookmark folders with anyone on the web with a personalised **share.opera.com** URL that's valid for 14 days.

One highlighted feature of the release is a new navigation bar that's now displayed at the bottom of the internal *Opera* pages namely the Speed Dial, the Bookmarks page and the Discover feature. The new bar adds consistency across these pages and lets the user switch between them with a single click.

The Print Preview window has also been restyled and lets you save web pages as PDF. The window gives you the option to print information about the web page, such as its URL, title and page number in the header and footer. Another useful feature is that you can turn off background graphics from pages before printing them.

The other major feature enhancement of this release is the return of tab previews. With the feature you can preview the contents of a tab without switching to it. You can do this by bringing up the list of open tabs by clicking on the tab list button in the tab bar and them hovering over them. Or, you can simply move the mouse over any inactive tab to bring up the preview. The previews are displayed in a large popup along with the title of the page open in the tab.

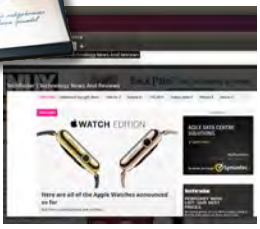
Business as usual

Although tab previews are a highlighted feature of this release, they aren't enabled by default. Even after enabling the feature, you have to open multiple tabs for the tab list button to appear. From then on the button is visible every time you launch the browser.

All said and done, while the new features are welcome additions, they aren't compelling enough to dump open source gems like *Firefox* and switch to the proprietary browser. There's nothing wrong with the browser as such, and we wouldn't write it off simply for being closed-source. The fact is that the open source alternatives, such as *Firefox* and *Chromium*, don't leave much to be desired.

In terms of features, *Opera* is largely similar to *Firefox*. Both have synchronisation features and can be extended by add-ons. The one feature unique to *Opera* is its Turbo mode which helps cut down bandwidth costs by compressing web pages before sending them to the user. This makes *Opera* a viable option for Android users with limited data plans, but isn't really useful for the average privacy-conscious desktop Linux user. LXF

Features at a glance



Tab previews

Click on the Expand Tabs button on the extreme-right of the tab bar presents a list of all open tabs.

Unified navigation

The Speed dial, bookmarks and discover pages now have a unified navigation bar.

LINUX FORMAT Verdict

Opera 27

Developer: Opera Software
Web: www.opera.com
Licence: Proprietary

Features	8/10
Performance	8/10
Ease of use	8/10
Documentation	8/10

» *Despite the high scores, we wouldn't recommend it because there are capable open source alternatives.*

Rating 8/10

Philips BDM4065UC

Is this the screen we've been searching for? **Jeremy Laird** thinks it might be.

Specs

- » **Size:** 40 inches
- » **Resolution:** 3,840 x 2,160
- » **Type:** VA
- » **Colour:** 8-bit
- » **Contrast:** 5,000:1
- » **Angles:** 178/178 degrees
- » **Response:** 8ms G-to-G (3ms with overdrive)
- » **Inputs:** DisplayPort, HDMI, MHL, VGA
- » **Stand:** Non-adjustable (200m VESA support)

› **Philips has adapted one of its 4K HDTVs to create a no frills 40-inch monitor.**

Even before the Philips BDM4065UC arrived at **LXF** Towers, we had a notion that when it came to 4K monitors, the HDTV market was going to be particularly relevant. HDTVs have always offered better value for money than monitors because they're cranked out in huge volumes. Conventional HDTVs though have made poor PC screens.

Meanwhile, the first budget 4K monitors have appeared, but offering just 28 inches, diagonally. Hardly titchy, but when you've got nearly 4,000 pixels across the horizontal, the dot pitch is tiny – and that generates all kinds of problems. Either you leave the scaling at 100% so that everything is eye-squintingly small, or you tweak the System Settings and browser scaling settings and everything looks hideous.

But what if you had a 4K PC monitor that was 40 inches in diagonal? Then you'd have a monitor on an epic scale with a pixel pitch in that workable window where it's fine enough to enable a crisp, sharp image, but not so fine you have to deal with the scale-or-not-to-scale dilemma. And guess what, 40 inches is into budget HDTV territory in terms of panels and production volumes. That explains why this massive Philips is cheaper than existing 30-inch 2,560 by 1,600 pixel monitors.

Hold on though, you can't just take a 4K HDTV off the shelf and use it as a monitor. That's because the bandwidth limitations of the current HDMI 1.4 standard restrict it to 30fps. What you need is the latest DisplayPort 1.2 interface, and you don't get that with TVs. Instead, Philips has taken a budget 4K HDTV model, ripped out the TV tuner, and bunged in a DisplayPort 1.2 interface to enable a 60Hz refresh from a single transport stream. No frills – just the minimum you need from a monitor.

That's great news. The main upsides are the affordability – okay, £600 is a lot of money, but spread over six, seven or more years and the annual cost looks reasonable. Then there's the slim shape of the chassis and the super-skinny bezel in traditional HDTV style.

Full of compromises

However, it's not all good news. The most obvious downside is a stand that doesn't adjust. At all. Though you do have the option of using the 200mm VESA mounts on the back to cook up something. Once you've seen something like *The Witcher 2: Assassins of Kings* running at 4K on a 40-inch panel, you're ruined for other screens.

Where the HDTV-ness kicks in with an unambiguous negative is the VA, or Vertical Alignment, rather than IPS panel tech. It delivers lovely deep blacks and nice, rich colours, but also viewing angles that are basically a bit broken. The problem is most obvious at the extremities of the display, especially along the bottom. If you sit back far enough, the issue resolves – but this is a PC monitor and the whole point is that you sit close. Is it a deal-breaker? Probably not, but it is enough to make us wonder whether this is only for early adopters willing to make compromises.

Of course, what we haven't mentioned is the sheer load any 4K panel puts on your graphics card; no single GPU is currently up to the job of running the latest games at 4K and full detail. Even multi-GPU setups with high-end cards will struggle. In mitigation, this screen looks decent running 1440p interpolated. So you could take the long term view – buy now and plan on a GPU upgrade in a couple of years – that should see you enjoying many years of native-res 4K gaming. LXF

Features at a glance

Display port

The only real way to drive this 4K display is viaa DisplayPort v1.2, while we wait for HDMI 2.0.

4K HDPI

The Linux desktop is ready for High Dots Per Inch displays. Notably, Ubuntu is well on its way..

LINUX FORMAT Verdict

Philips BDM4065UC

Developer: Philips
Web: www.philips.co.uk
Price: £608

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

» *Cheaper than many 30-inch monitors, but with vertical alignment issues and quite a hefty price tag.*

Rating 8/10

Dying Light

Skipping and hopping **Christopher Livingston** is parkouring his way to zombie-free safety.

Specs

- **OS:** Ubuntu 14.04 or Steam OS
- **CPU:** Intel Core i5-2500 or AMD FX-8320
- **Memory**: 4GB DDR3
- **HDD:** 40GB
- **GPU:** Nvidia GeForce GTX 560 (1GB VRAM)

Like its hero, *Dying Light* has its ups and downs and is kind of all over the map. Techland, creator of the *Dead Island* series, takes elements from a number of games and mashes them together in its open world first-person zombie shooter. It's an uneven mix, bookended by a slow start and an exasperating finish, not to mention a few troubling performance issues, but in the middle lies a sweet spot that provides hours of satisfying, zombie-stomping fun.

Our hero, Crane, could be mistaken for the Inspector Clueso of the secret-agent world, managing to skydive into a locked-down zombie-filled city in broad daylight, immediately get accosted by three thugs, gets infected and also a friendly killed. All before the game starts. Despite being what we assumed was a top physical specimen, he initially can't run for long before slowing and gasping, and can only swing a melee weapon a few times before running out of stamina.

Weapons, at first, are limited to pipes, small knives, or sticks. Blueprints let you craft upgrades to electrify pipes, sharpen or poison knives, add nails to a baseball bats. Better weapons like swords and axes slowly begin to appear as you progress, while it'll be hours of play before you get a handgun.

So your best weapon is Crane's slowly improving agility coupled with a city perfectly built for climbing and roof-running. Once you get the hang of leaping and climbing, *Dying Light* opens up and becomes a fun, zombie-infested playground. Techland has done a great job with the running, jumping, climbing, and clambering. Skill points are slowly doled out and there's lots to spend them on: vaulting to jump over zombies, flying kicks and sliding kicks are the norm, exotic stealth and grappling hook skills take a lot more gameplay to attain. It's a well paced steady climb in improving abilities.

This is my club, there are many like it, but this one is mine.

You might want to get that lump looked at.

Right, zombies!

Mostly, they're slow, shambling types, clogging the surface streets and occasionally comedically toppling off rooftops. There are also specials: huge brutes who swing clubs, spitters who barf slime, and bloated blobs who scream then explode.

When night falls, the zombie game changes entirely. A new breed of zombie called Volatiles appear, and suddenly Crane is no longer the fastest thing on two legs in Harran. Wander into their vision cone and they pursue you at a lightning-fast pace while screeching to attract other zombies.

Harran itself is peppered with stuff to do, much of it familiar from other open world games. In addition to lengthy story missions, there are multipart side-quests, looting and scavenging expeditions, securing safe-houses, and a few timed challenges. The story itself, it's a bit of a clunker as Crane, supposedly torn between his loyalties, grapples unconvincingly with his conscience despite very obviously being a complete Boy Scout.

Considering the formulaic story and the uninteresting characters, we appreciate being able to skip cutscenes and speed up conversations. That said it also contains classic gaffs like a QTE knife fight, a couple of deathmatch arena fights with no weapons, and a tedious lengthy dream sequence.

When it wasn't tripping over its own feet, be it technical or the story, *Dying Light* was really enjoyable. We came away having had a good time. There are frustrations here, but there's also an exciting movement system and a healthy list of activities to engage in.

The Linux release experienced serious early issues, largely on AMD Radeon cards and owners should wait for suitable patches. Even Nvidia cards had performance issues, but turning off certain shadows and DoF boosted frame rates. Techland has already said that it's working on patches, which should now be available. LXF

LINUX FORMAT Verdict

Dying Light

Developer: Techland
Web: dyinglightgame.com
Price: £39.99

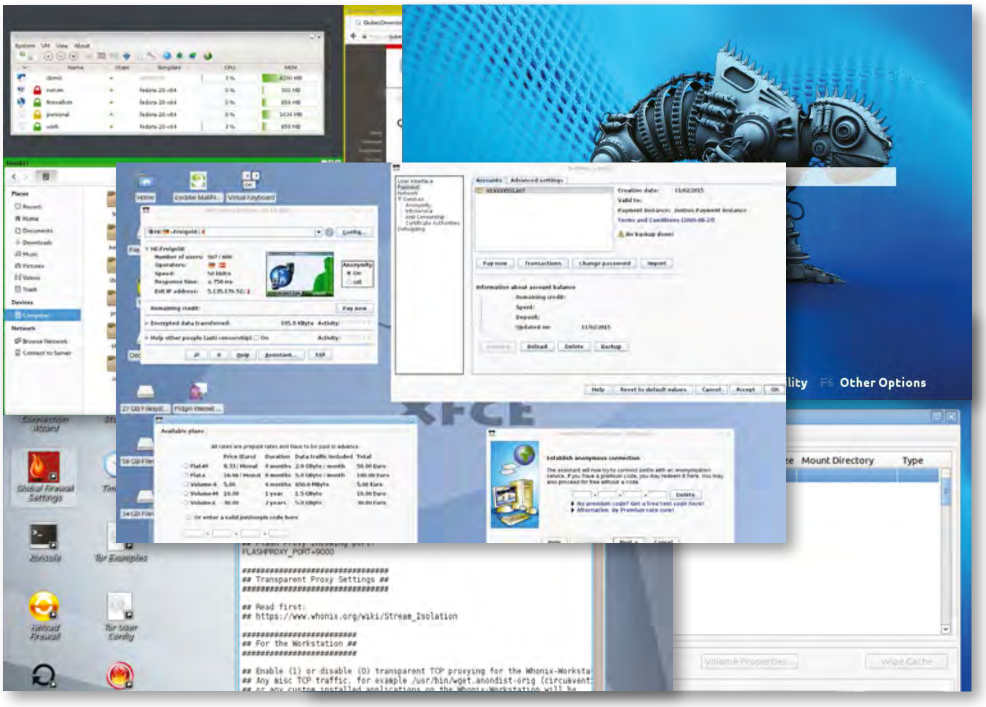
Gameplay	9/10
Graphics	8/10
Longevity	7/10
Value for money	7/10

A vastly fun first-person zombie shooter that stumbles on its story and some big technical issues.

Rating 8/10

Privacy distros

Cover your tracks and keep your identity private. **Alexander Tolstoy** compares special-purpose Linux distros to help you stay invisible on the web.



```
FLASHPROXY_PORT=9000

#################################
## Transparent Proxy Settings ##
#################################

## Read first:
## https://www.whonix.org/wiki/Stream_Isolation

##########################
## For the Workstation ##
##########################

## Enable (1) or disable (0) transparent TCP proxying for the Whonix-Worksta
## Any misc TCP traffic. for example /usr/bin/wget.anondist-orig (circumvent
```

How we tested...

Nearly two years ago mainstream media started discussing PRISM, which raised a lot of concerns about privacy and anonymous access to the Internet. Shortly after that *Linux Format* came out with great Anonymous distros round-up [*see Roundup, p26,* **LXF174**], which highlighted a noticeable outburst of new releases for Tails, Whonix and other Linux distributions for the security conscious user. This time we revisit the topic with a different selection of contenders and a changed perspective, too. We'll cover: the current state of actively maintained distros; their availability; ease of use; performance; feature set and documentation, and last, but not least; we'll cover the level of compromise they require for regular, general-purpose computing.

Our selection

- » JonDo Live-DVD
- » Qubes OS
- » Tails
- » Ubuntu Privacy Remix
- » Whonix

There are numerous use cases where someone security conscious may want to use a specialised and non-mainstream Linux distribution instead of a regular one. So we selected five diverse options, each with its own traits and benefits.

Tails is perhaps the most well-established system we're covering, and claims to provide anonymous internet access, circumventing any censorship. Ubuntu Privacy Remix (UPR) provides anonymity together with a strong means of securing your data. It runs only in live mode, encrypts your data and protects it against unsolicited access. Whonix boasts nearly the same features as Tails but goes even further by dividing your workflow into two parts: server and workstation. Qubes OS implements the 'security by compartmentalisation' approach [*see Reviews, p28,* **LXF164**], but this time will face off against other alternatives. Finally, JonDo Live-DVD is a very interesting solution, which grew out of the multiplatform JonDonym, an internet surfing anonymiser with privacy and security in mind.

> **"The winner should be not only secure, but balanced and friendly even to less tech-savvy users."**

Anonymity and security tend to go hand in hand, so we expect an added benefit to be being able to nail down your system so it's secure from would-be hackers. We'll compare all these options with each other in different terms, and the winner should be not only secure, but generally balanced and friendly even to less tech-savvy users.

Availability

What does it take to get them running?

When you decide to try out an anonymous distro, you have to be aware that there's cost involved in using them, but it varies, so lets see what it takes to get our contenders up and running.

Tails is the most well-known distro, and we expected to download its ISO file and write it onto USB stick via some convenient tool like *dd* or front-end like *ImageWriter*. But the process with Tails turns out to be less straightforward, because the image has to be modified with the *isohybrid* utility. So, it went:

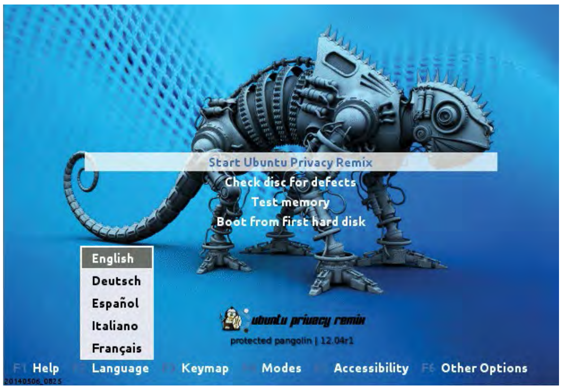
```
isohybrid tails-i386-1.2.3.iso -h 255 -s 63
dd if=tails-i386-1.2.3.iso of=/dev/sdc bs=16M
```

Where **/dev/sdc** is your flash drive. After that it works like a charm. The system boots into the live session just like a regular Debian-based distro.

Whonix and Qubes OS are significantly harder to launch, and here is why: Whonix comes in the form of two *Virtualbox* machines, one for the Gateway and another for the Workstation. The idea behind this exquisite delivery is to isolate the environment you work in from the internet access point. So, the first thing to do is launch and configure the Whonix Gateway on one VM and then accessing it from another VM, where all work will be done. We didn't find any issues with it, but we have to admit that only advanced users will be able to deploy their workflow under Whonix.

After writing Qubes OS's ISO onto USB stick and booting from it, we discovered that there's no live session, only an installation mode. Qubes OS is based on a recent Fedora release and shares the same installer with it. But the system has some quite surprising system requirements: it wants you to provide it with 4GB of RAM, 32GB for the root partition and prefers built-in Intel video chip, as Nvidia or AMD have some issues in Qubes OS. The system needs such overstated resources due to its 'Security via isolation' approach, which we'll discuss later.

Finally, Ubuntu Privacy Remix and JonDo Live-DVD were extremely easy to launch. Their respective live sessions were fast and easy to use.



› No, it's not a blue SUSE lizard, it's Ubuntu Privacy Remix, which features this cool Protected Pangolin!

Verdict

JonDo Live
★★★★★

Qubes OS
★★★☆☆

Ubuntu Privacy Remix
★★★★★

Tails
★★★☆☆

Whonix
★★☆☆☆

» *Easy access to anonymous live sessions wins out.*

Development state

Private and secure today, but how actively are they maintained?

This aspect is often overlooked, but it's vital as regular users will want to have an up-to-date and actively supported distro. The reality is that some secretive distros are abandoned by developers (such as Privatix) or left unmaintained for years (like Liberté). Some may think that it's a matter of new features and fixes, but let's not forget that abandoned Linux distros may have trouble running on modern hardware that has things like UEFI and Secure Boot.

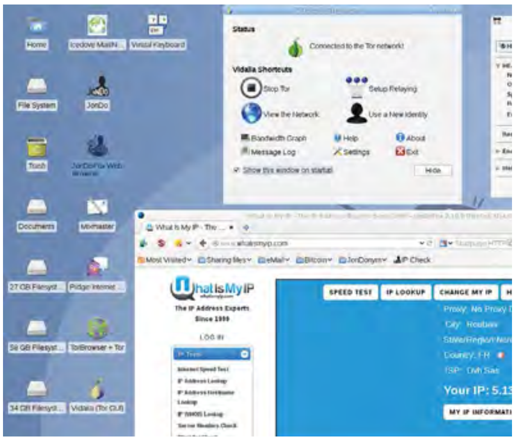
Tails is one of the best maintained security distros, with a very fast pace of development. New releases are rolled out every 2-4 months, which means Tails has had six releases during 2014 and went from v0.23 to 1.2.3 rapidly.

The Ubuntu Privacy Remix (UPR) developers, in comparison, don't seem to be in such a hurry, but keep development steady. UPR emerged in December 2008 and has been sticking with Ubuntu LTS releases. The current version is 12.04r1 (Protected Pangolin) which supports new hardware but is still a very lightweight distro.

Whonix is a relatively new project, which started in 2012 and has been very actively developed since then. Now at version 9.6, Whonix continues to get updates every few months.

Qubes OS is similar in that its birth also dates back to 2012, and the project has reached R2 release. Qubes OS's development is very active, with lots of well-documented alpha, beta and release candidate versions published every few months.

But that leaves us with the insanely speedy development record of JonDo Live-DVD. Somewhat staggeringly, JonDo boasts a changelog, which is updated every 5-10 days!



› JonDo Live-DVD has embarassingly frequent updates.

Verdict

JonDo Live
★★★★★

Qubes OS
★★★★★

Ubuntu Privacy Remix
★★★★☆

Tails
★★★★★

Whonix
★★★★★

» *All our participants are in rude health & updated often.*

Web surfing protection

How effectively do they shield you from web threats?

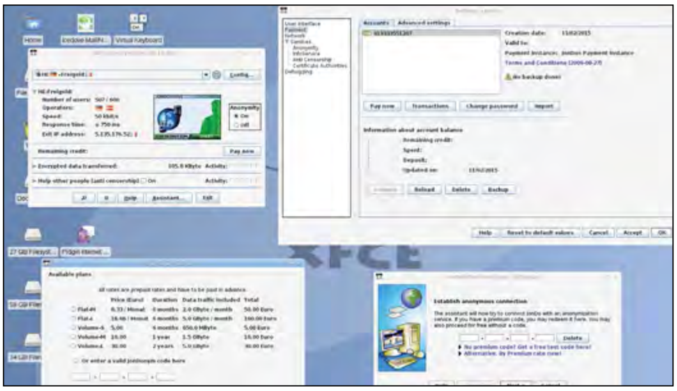
When you're accessing the internet, things become complicated and no one can guarantee that everything you access is 'absolutely' safe. But most of our distros try their best to offer the maximum possible protection.

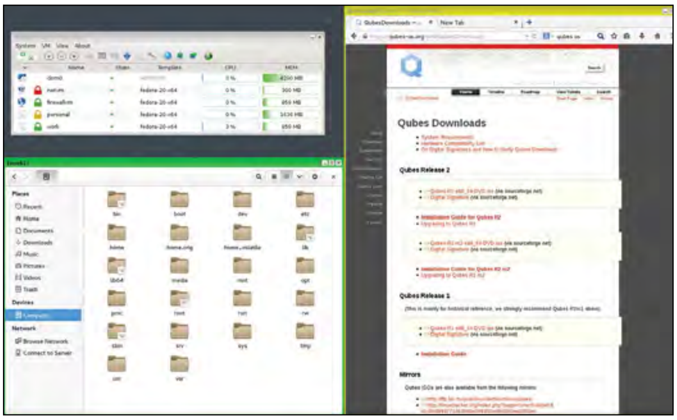
We also assume that while security is a top priority, users will still need to: access webmail; download and upload files; store passwords and sensitive data; and perform other common activities on the internet. Anonymity requires some compromises, such as lower download speeds and a harder password policy, but we also insist on a comfortable web browsing experience. But don't confuse greater security and hardened internet policies with good user data safety. This is different and something we'll cover later.

JonDo Live-DVD ★★★★★

JonDo provides network anonymity using the JonDo *IP changerv* (aka *JonDonym*), which is a Java Anon Proxy, similar to *Tor*. JonDo enables web browsing (via a *Firefox*-based *JonDoBrowser*) with revocable pseudonymity and sends requests through a cascade and mixes the data streams of multiple users to further hide the data to outsiders.

It's worth noting that while the whole thing is open source, there are free and commercial plans. The free one can only use destination ports 80 and 443 that are used for the HTTP and HTTPS protocol (enough for web browsing and FTP). The premium service provides additional SOCKS proxies for extra anonymisation and a better connection speed. Generally, we find JonDo safer than Tor, because JonDo is much more centralised and can't include malicious nodes (which is possible in Tor).





Qubes OS ★★★★☆

Qubes OS implements another concept of virtualisation-based isolation. The system runs *Xen* hypervisor with multiple instances of an altered Fedora 20 virtualised on top of it. Qubes OS is divided into several 'domains' and applications can be run as virtual machines (AppVMs).

The standard way of anonymising network traffic is using Qubes TorVM, which connects to the internet and runs Tor. Other applications can be assigned to use this 'Torified' connection. The positive side is that an application doesn't need to be aware of Tor; it runs in regular mode without needing add-ons, and all IPv4 TCP and DNS traffic is routed by Tor. The downside is that you need to configure everything manually. We also noticed that this concept tends to restrain attacks and malware from spreading outside domain/AppVM, rather than prevent them.

Data safety

How safe is your sensitive data within each distro?

Though the most important feature of Tails is its 'amnesia' in live mode, you can install it to your hard drive and use it just like a regular Linux distro. Among all of the benefits of doing that, you'll note that your RAM will be wiped on reboot or shutdown, which will protect against forensic recovery techniques.

Ubuntu Privacy Remix shines when it comes to securing your data. The only way to store it is using the extended *TrueCrypt-Volumes*, which can be stored on removable USB media only (which, in turn, is mounted with a 'noexec' option). There's no way for your data to be left on drive partitions, not even unnoticed or by accident.

Whonix is much less amnesic than most of the others. On the Workstation side all data can be stored persistently, and its up to you how you keep it. You may want to encrypt and protect it with an extra password or store it on isolated location. But generally Whonix doesn't have a strong focus on data security.

Qubes OS is much better for data security, because it's possible to isolate sensitive data in a separate domain/AppVM without network access, but again the security level is heavily dependent on the skill of the user and how disciplined they are. JonDo Live-DVD offers a way for using persistent storage, and we found it to be quite user-friendly. It's ready to use LUKS encrypted USB sticks and drives and provides a special assistant to prepare your media.

Verdict

JonDo Live ★★★★☆

Qubes OS ★★★★☆

Ubuntu Privacy Remix ★★★★★

Tails ★★★★☆

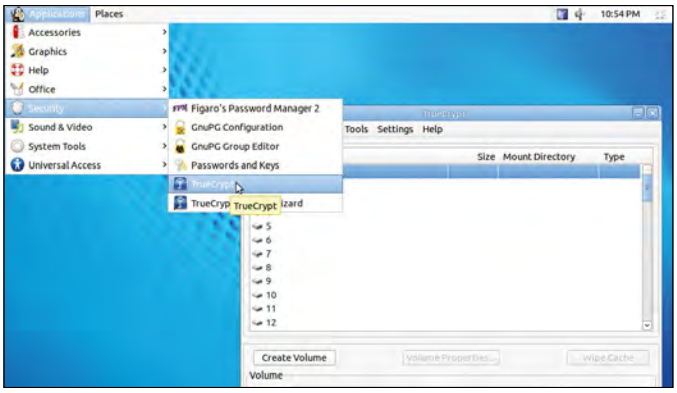
Whonix ★★★☆☆

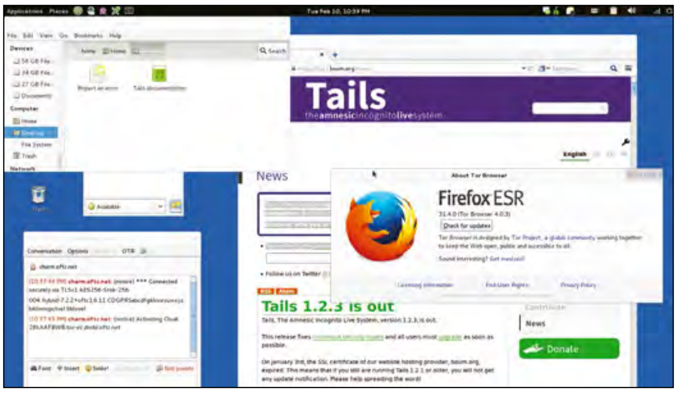
» *This time UPR offers the most security for your data.*

Ubuntu Privacy Remix ★☆☆☆☆

Sad but true, Ubuntu Privacy Remix (UPR) has no networking functionality at all. The system kernel is modified so that it ignores any network hardware, making UPR a perfectly isolated system, which can't be attacked via LAN, WLAN, Bluetooth and Infrared etc. So, there's no web browsing, no cookies, no trojans nor any data downloaded from the web, and no instant messaging or remote or cloud services. Almost all traces of network connectivity are wiped off the UPR, though some are still there. For example, *ifconfig* and *ifup/ifdown* commands are there, but they are virtually helpless, as network hardware is violently disabled.

So in this test UPR fails to be any use for web surfing, even if it is part of the design. If, however, you're paranoid and want a system that avoids being online entirely then UPR will be the right solution.





Tails 1.2.3 is out

Tails, The Amnesic Incognito Live System, version 1.2.3, is out.

This release fixes numerous security issues and all users must upgrade as soon as possible.

On January 3rd, the SSL certificate of our website hosting provider, boum.org, expired. This means that if you still are running Tails 1.2.1 or older, you will not get any update notification. Please help spreading the word!

Tails ★★★☆☆

Tails includes top-notch networking features, and the most important one is Tor, which is an open network of anonymous servers that attempts to prevent your identification and traffic analysis.

This is accompanied by *Vidalia*, a front-end for easy set up, a preconfigured *Firefox* ESR-based web browser, which is equipped with a Tor Button, *HTTPS Everywhere, NoScript* and *AdBlock Plus* extensions.

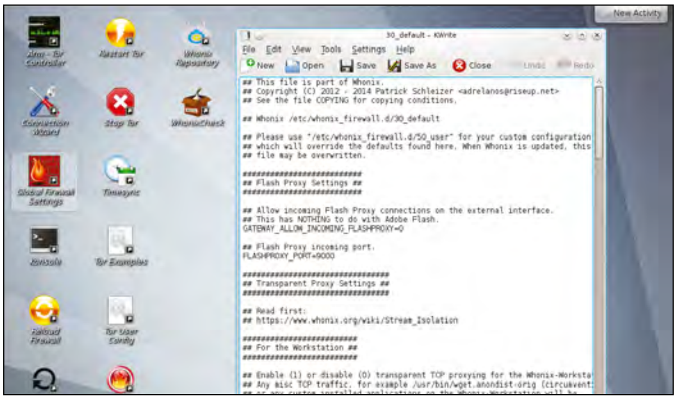
Tails many extras include *I2P* anonymising network, proxy and VPN front-ends, the *Florence* virtual keyboard, application isolation via *AppArmor*, *PWGen* for generating strong passwords and *KeePassX* for managing them, and *AirCrackNG* for wireless networks auditing etc.

Tor and I2P traffic are also divided, thanks to the dedicated *I2P Browser*, and *Pidgin* uses the more secure Off-the-Record (OTR) mode.

Whonix ★★★★★

Whonix also relies on Tor for network anonymity and shares many third-party tools with Tails. So lets point out the differences. Here the Tor client runs on Whonix-Gateway, which provides better protection against IP and location discovery on the Workstation.

The level of IP and DNS protocol leak protection is sometimes the same, but in Tails there's a possibility of misconfiguration, which can lead to IP leak and in Whonix this doesn't exist. Even if the workstation is compromised (eg by someone getting root access), it would still be impossible to find out the real IP. Isolating the proxy server within a standalone VM (or maybe a physical PC) works great. Whonix also makes use of 'entry guards' in Tor (randomising endpoints), which is something that is missing in Tails out of the box.



```
## This file is part of Whonix.
## Copyright (C) 2012 - 2014 Patrick Schleizer <adrelanos@riseup.net>
## See the file COPYING for copying conditions.

## Whonix /etc/whonix_firewall.d/30_default

## Please use "/etc/whonix_firewall.d/50_user" for your custom configuration
## which will override the defaults found here. When Whonix is updated, this
## file may be overwritten.

###########################
## Flash Proxy Settings ##
###########################

## Allow incoming Flash Proxy connections on the external interface.
## This has NOTHING to do with Adobe Flash.
GATEWAY_ALLOW_INCOMING_FLASHPROXY=0

## Flash Proxy incoming port.
FLASHPROXY_PORT=9000

#################################
## Transparent Proxy Settings ##
#################################

## Read first:
## https://www.whonix.org/wiki/Stream_Isolation

##########################
## For the Workstation ##
##########################

## Enable (1) or disable (0) transparent TCP proxying for the Whonix-Worksta
## Any misc TCP traffic, for example /usr/bin/wget.anondist-orig (circumvent
```

Performance

How snappily do they run?

More recent Tails uses 3.16.7 kernel and loads into Gnome Shell 3.4 in fallback mode by default. The desktop is very lightweight; nearly as fast as classic Gnome 2 in previous Tails releases, but official system requirements say it needs at least 1GB of RAM to work smoothly, which we think is a bit much.

Ubuntu Privacy Remix was updated to use the Ubuntu 12.04 LTS package base and thus has numerous backports and modern features, yet it remains very easy on resources. UPR uses a classic Gnome 2 desktop, which loads in a couple of seconds. We'd suggest that 512MB of RAM is enough, though UPR can make use of the larger RAM volume as the system implements 'ramzswap' to store swap file in RAM.

JonDo Live-DVD can boot even on very old CPUs, and its XFCE desktop is very fast. However, you'll need 1GB RAM to work smoothly with the Java-based JonDo app and the web browsers.

Whonix is different, again, because you need a host capable of running two *Virtualbox* guest machines at a time. Your host OS and configuration is down to you, but you're going to need at least 4GB of RAM, a spare 12GB of hard drive space. However, the SSD and CPU with hardware virtualisation support are both very welcome.

For Qubes OS you'll need an even beefier machine: a 64-bit CPU, 4GB of RAM and at least 32GB for root partition. Qubes OS is, therefore, the most demanding choice.

Verdict

JonDo Live
★★★★★

Qubes OS
★★☆☆☆

Ubuntu Privacy Remix
★★★★★

Tails
★★★★☆

Whonix
★★☆☆☆

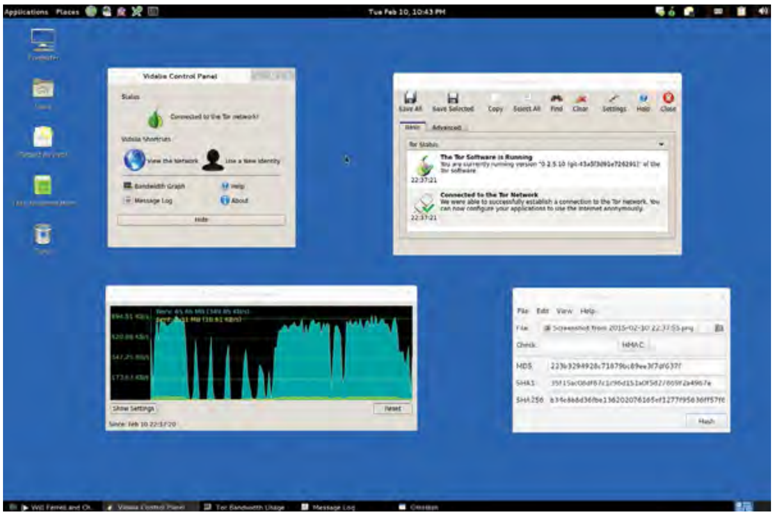
» *Both Tails and JonDo are modest on resources.*

Desktop usability

Can you be anonymous and still enjoy a feature-rich desktop?

Though Tails is 'amnesic', it includes an installer, which can create a persistent partition either on the same USB stick you boot from, or another USB storage device. This makes Tails a pleasant experience for permanent work in live mode. It also includes a vast selection of software, from *LibreOffice* and *Gimp* to *Audacity* and *Sound Juicer*.

JonDo Live-DVD also has a very usable Xfce live desktop, which is packed with all the essential desktop software, but its main advantage is that you can install both the JonDo IP changer and *JonDoFox* browser on any Linux distro. This is a huge bonus, because you can stay with your already-configured Linux box and seamlessly turn anonymous.



Connected to the Tor network!

The Tor Software is Running
You are currently running version "0.2.5.10 (git-...)" of the Tor software
22:37:21

Connected to the Tor Network
We were able to successfully establish a connection to the Tor network. You can now configure your applications to use the Internet anonymously.
22:37:21

Since: Feb 10 22:37:20

File: Screenshot from 2015-02-10 22:37:55.png

❯ **The desktop in Tails will be familiar and easy to use for Gnome users.**

Ubuntu Privacy Remix (UPR) includes only basic Gnome 2 accessories and very few desktop apps (*Scribus* and *LibreOffice* are the most noticeable examples). The desktop experience in UPR is poor, so much so that even extracting screenshots turned out to be a problem. Worst of all, UPR is made deliberately non-manipulative, so nothing can be fixed from a desktop perspective.

Both Whonix guest machines use the KDE desktop on top of Debian. We really love KDE, but it seems to be excessive on the Gateway side. But the Workstation experience turned out to be very comfortable. Aside from some minor slowdowns and restrictions, because of it being a virtualised and firewalled system, Whonix Workstation can be used as a fully featured desktop.

Qubes OS is an entirely different experience: it's easy to install but can work very slowly later down the line. Its KDE desktop is intuitive, but interaction between domains requires extra skill. For example, copying and sharing files from one domain or AppVM to another has its own logic and clipboard usage is limited.

Verdict

JonDo Live
★★★★★

Qubes OS
★★☆☆☆

Ubuntu Privacy Remix
★★★★★

Tails
★★★★☆

Whonix
★★☆☆☆

» *The best offer familiar software and anonymity tools.*

Documentation and support

Is there any help and where do you get answers to questions?

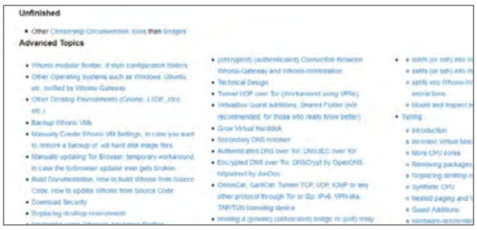
Good wiki pages, FAQs and other helpful documentation are important for any software. This is certainly the case with anonymous distros that can be frustrating even for people familiar with Linux.

Tails offers in-depth end-user documentation with general information, first steps, commonly asked questions and detailed explanations for almost all aspects, even those not related to Tails directly, but it's all essential if you want to study the basics of privacy and encryption. There's even a chat room and a 'request a feature' form.

Ubuntu Privacy Remix has a neat and compact website, yet there isn't that much materials, but the quantity of UPR resources corresponds with its feature set. You can find some helpful how-to guides, such as instructions for creating a personal UPR build (with a custom software set).

Nearly all Whonix documentation resides in a dedicated and detailed wiki portal. We found it to be very comprehensive and more in-depth than the resources Tails supplies – Whonix has more articles, more support options and a very active forum.

The Qubes OS project also has a wiki portal with essential and advanced articles. The OS architecture is explained in detail and there's an FAQ, tutorial slides and user documentation. Qubes OS has many extra features, such as running non-Linux AppVMs, and this is covered in a detailed manual.



Unfinished

- Other Censorship Circumvention Tools than Bridges

Advanced Topics

- Whonix modular flexible .d style configuration folders
- Other Operating Systems such as Windows, Ubuntu, etc. torified by Whonix-Gateway
- Other Desktop Environments (Gnome, LXDE, Xfce, etc.)
- Backup Whonix VMs
- Manually Create Whonix VM Settings, in case you want to restore a backup of .vdi hard disk image files
- Manually updating Tor Browser, temporary workaround, in case the torbrowser updater ever gets broken
- Build Documentation, How to build Whonix from Source Code, How to update Whonix from Source Code
- Download Security
- Replacing desktop environment
- (encrypted) (authenticated) Connection Between Whonix-Gateway and Whonix-Workstation
- Technical Design
- Tunnel UDP over Tor (Workaround using VPNs)
- VirtualBox Guest Additions, Shared Folder (not recommended, for those who really know better)
- Grow Virtual Harddisk
- Secondary DNS resolver
- Authenticated DNS over Tor, DNSSEC over Tor
- Encrypted DNS over Tor, DNSCrypt by OpenDNS, httpsdnsd by JonDos
- OnionCat, GarliCat: Tunnel TCP, UDP, ICMP or any other protocol through Tor or i2p; IPv6, VPN-like, TAP/TUN tunneling device
- Hosting a (private) (obfuscated) bridge or (exit) relay
- sshfs (or ssh) into
- sshfs (or ssh) into
- sshfs into Whonix-W instructions
- Mount and inspect i
- Tuning
  - Introduction
  - Increase Virtual Mac
  - More CPU cores
  - Removing packages
  - Replacing desktop
  - Synthetic CPU
  - Nested paging and V
  - Guest Additions
  - Hardware-accelerat

❯ **The Whonix help section is huge and scrollable. Even advanced and in-depth topics are covered.**

There's also a helpful developer's corner, which provides all you need to develop custom solutions.

JonDo has help topics, an FAQ, tutorials, a wiki portal and a forum. Though it looks complete, a thorough review shows many weaknesses. The FAQ is brief, and the wiki is very small. Very few topics are actually covered, which is disappointing.

Verdict

JonDo Live
★★★☆☆

Qubes OS
★★★★☆

Ubuntu Privacy Remix
★★★☆☆

Tails
★★★★☆

Whonix
★★★★★

» *Whonix sneaks in front of Tails for it's level of support.*

Privacy distributions

The verdict

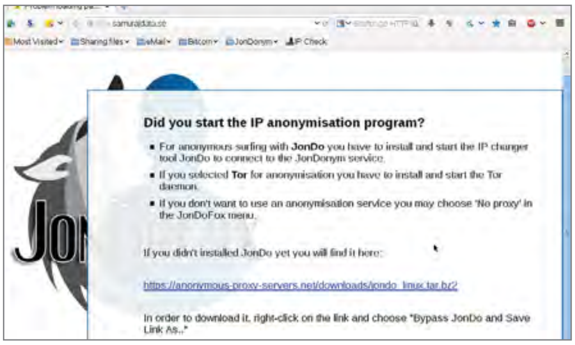
Java Anon Proxy was a 2007 startup, backed by solid research work of many years. Here, we witness the fruit of that work as JonDo Live-DVD clearly outperforms the former king of anonymous web access: Tails. Both projects are premiere quality, however, with balanced features and active development.

It's hard to say whether Tor provides perfect anonymity or not, but it's technically possible to single out a Tor user either through a compromised node or by matching traffic and user behaviour with other details, or even by correlation-timing attacks. On the other hand, JonDo node selection is less random than Tor, and we're not completely sure to what extent you can trust it. Both solutions slow the internet speeds greatly, and the JonDo proxy cascade seems to be even slower than Tor node chain. But connection speed is not top priority, because you're getting well-tested and supported anonymity.

Other participants clearly defined the cost they charge for advanced privacy and security. Whonix forces you to use virtual machine, which is always slower than a host computer, has little or no 3D support and takes extra time and skills to install it for the first time. But once you've done that Whonix can be configured to your need just like any other Debian-based distro.

It would also appear that Qubes OS will only work on quite high specified hardware, but even then it runs even slower than virtualised Whonix. Qubes OS does, however, deliver good anonymity, but its main purpose is to isolate different segments so that one segment can't bring down the others if compromised. You will also have to learn how different software domains communicate with each other.

The approach of Ubuntu Privacy Remix is unconventional, but it's also about anonymity although dealing with it very differently to the others. The project's website shows how you can create your own UPR spin-off and use it as a perfectly isolated system, which leaves no traces on a computer. UPR can also detect virtual environments and eject its ISO from its settings, but all this is solely local, without any connectivity with the outside world.



Did you start the IP anonymisation program?

- For anonymous surfing with **JonDo** you have to install and start the IP changer tool JonDo to connect to the JonDonym service.
- If you selected **Tor** for anonymisation you have to install and start the Tor daemon.
- If you don't want to use an anonymisation service you may choose 'No proxy' in the JonDoFox menu.

If you didn't installed JonDo yet you will find it here:

https://anonymous-proxy-servers.net/downloads/jondo_linux.tar.bz2

In order to download it, right-click on the link and choose "Bypass JonDo and Save Link As.."

› *JonDoFox* won't let you surf the internet unless your start Java Anon Proxy.

> "JonDo Live-DVD clearly outperforms the former king of anonymous web access: Tails."

1st JonDo Live-DVD ★★★★★

Web: http://bit.ly/JonDoLive-DVD **Licence:** BSD **Version:** 0.9.71.2

» Fast, portable, effective and easy to use for anonymous web surfing.

2nd Tails ★★★★★

Web: https://tails.boum.org **Licence:** GNU GPLv3 **Version:** 1.2.3

» Balanced for 'mostly' safe internet access. Also a friendly way to try Tor.

3rd Whonix ★★★★☆

Web: www.whonix.org **Licence:** Mainly GNU GPL **Version:** 9.6

» Very usable and super-secure, but the hardware specs are quite high.

4th Qubes OS ★★★☆☆

Web: https://qubes-os.org **Licence:** Mainly GNU GPL **Version:** R2

» Very secure, but like riding a bumpy narrow road between concrete walls.

5th UPR ★★☆☆☆

Web: www.privacy-cd.org **Licence:** Mainly GNU GPL **Version:** 12.04r1

» Consider it as a special-purpose distro for securing sensitive data.

Over to you...

Tell us about your anonymous web surfing experiences at **lxf.letters@futurenet.com**. What's your favoured distro for privacy?

Also consider...

Many people share the illusion that they can be invisible and unreachable under the Tor network. In fact, this is only true until a user breaks a law or somehow attracts attention from intelligence services. Please use anonymity only for peaceful purposes and at your own risk. On the other hand, you have a right to keep your data away from third-parties, so why not take some measures?

The choice of anonymising distros is larger than what we've covered. Privatix and Liberté both haven't received any updates for a long time, but they are still usable and ready for web surfing on most machines. There are other projects too, such as IprediaOS, Polippix and Mandragora that didn't fit in this Roundup but are worth considering. In fact, it's not too hard to turn your existing Linux install into a digital fortress. Almost all tools for anonymity on Linux are open source, including Tor front-ends, extensions and encryption methods. LXF

Subscribe to LINUX FORMAT

Get into Linux today!

Read what matters to you when and where you want.
Whether you want Linux Format delivered to your door, your device, or both each month, we have three great options to choose from.*

Choose your LINUX FORMAT package today!

#1 for Free Software

Print

FREE DVD!
UBUNTU + FEDORA UPDATED!
LINUX FORMAT
Get into Linux today!
GET STARTED WITH LINUX!
Beginner's guide
» Try Linux instantly
» Dual-boot Windows
» FREE full software!
PLUS The top three distros explored!
RASPBERRY Pi 2!
500% FASTER
EXCLUSIVE! Hands-on guide and in-depth review of the all-new Pi 2
IS YOUR FREE DVD MISSING? Speak to your newsagent now!
Made of Wynn
'Optimising for happiness' is kind of key to the Ruby community
Wynn Netherland on looking after GitHub p44
Switch to Python 3!
» Start programming today and easily port your code
Blogging
Get into Ghost
» How to install the next-gen blog platform
Video streaming
Linux Motion
» Build a secure home CCTV system
TuxRadar.com

Every issue delivered to your door with a 4GB DVD packed full of the hottest distros, app, games and more. **PLUS** exclusive access to the *Linux Format* subscribers-only area.

ONLY £31.99

Your subscription will then continue at £31.99 every 6 months – SAVING 17% on the shop price.

Digital

Instant access to the digital editions of the magazine on your iPad, iPhone and Android* devices. **PLUS** exclusive access to the *Linux Format* subscribers-only area, featuring complete issues & disc downloads.

ONLY £20.49

Your subscription will then continue at £20.49 every 6 months – SAVING up to 37% on the shop price.

*Only available in certain territories: http://bit.ly/LXFwhere

Get the complete LINUX FORMAT package

Print + Digital

Now on Android!

» A DVD packed with the best new distros and free & open source software every issue.

» Exclusive access to the *Linux Format* archive – with 1,000s of DRM-free tutorials, features, and reviews.

» Every new issue of the magazine in print and on iPad, iPhone, and Android* devices.

» Never miss an issue, with delivery to your door and straight to your device.

» Huge savings, the best value for money, and a money-back guarantee.

ONLY **£38.49**

Your subscription will then continue at £38.49 every 6 months – SAVING 17% on the shop price and giving you up to a 78% discount on a digital subscription.

Two easy ways to subscribe...

Online: myfavouritemagazines.co.uk/LINsubs

Or call 0844 848 2852

(please quote PRINT15, DIGITAL15, BUNDLE15)

Prices and savings quoted are compared to buying full-priced UK print and digital issues. You will receive 13 issues in a year. If you are dissatisfied in any way you can write to us or call us to cancel your subscription at any time and we will refund you for all undelivered issues. Prices correct at point of print and subject to change. For full terms and conditions please visit: **myfavm.ag/magterms**. Offer ends 19/04/2015

TOP 100 LINUX TOOLS

Take a stroll through the open source garden with **Mayank Sharma** as he picks the best apps, tools and utilities available to all Linux kind.

With 70 Raspberry Pi top apps!

We all have our favourite open source apps that work for us better than any available alternative. But take a moment and step back from the *Emacs* vs *vim* type battles raging on in the Linux-verse and marvel at the sheer number of apps at our disposal. Your distros' software repositories give you access to thousands of apps, and you can install everything from fully featured app suites to nifty command-line utilities literally with the touch of a button.

There are open source apps and tools for all kinds of applications today. There's hardly any use case that isn't catered for by a community contributed app. Many of these apps have proved their mettle and offer features and performance benefits that surpass their proprietary counterparts. They have also proved themselves to be invaluable to home and business users in more than one sense of the word. According to rough estimates on **www.openhub.net**, some popular apps such as *LibreOffice*, *Firefox* and *Apache* would take several hundred person-years to develop and cost millions of pounds. Yet they are all available to you for no-cost.

> "Many of these apps have proved their mettle and surpass their proprietary counterparts."

Open source apps come in many shapes and sizes and you can grade them based on their usability. There are feature-rich apps, task-oriented app suites, well put-together tools, and newfangled novelty apps and games. Some ship with well-designed graphical interfaces and others are more versatile when operated from the command-line.

In this feature, we traverse this diverse and vast collection of open source gems on offer and pick the ones that are at the top of their game. In this list of the 100 best apps we've covered a wide range of categories. Whether you are a business owner, an educational institution, a developer, a home user, or a gamer, we've got something for everyone. While you'll be familiar with some of the most popular tools in this list, rest assured there are quite a few that might have missed your attention. If you've been unable to escape the clutches of commercial software, we're sure you'll find quite a few tools on this list that are suitable replacements.

Essential apps

A Linux desktop isn't complete without them.

LibreOffice

Forked from *OpenOffice.org*, *LibreOffice* has become one of the most popular office productivity suites. It includes programs for word processing, and can create spreadsheets, slide shows, diagrams and drawings, maintain databases, and compose mathematical formulae. It also offers good compatibility with documents in proprietary formats and has recently had a face lift.

www.libreoffice.org

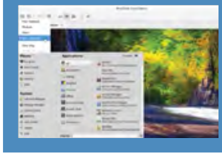
Wine

Despite the increasing number of cross-platform apps that work on Linux, there are some that still only support Windows. This includes big third-party proprietary apps, such as *Adobe Photoshop* or just small niche home-grown tools that you can't do without. For such situations, you can use *Wine*, which generally run these Windows-only apps and games with ease. The project supports over 20,000 apps. Some work flawlessly out-of-the-box while others require minor configuration tweaks.

www.winehq.org

Remmina

With *Remmina* you can access a remote computer from the comforts of your desktop. It supports the widest range of protocols and will connect to all kinds of remote desktop servers. The app is easy to use, and has enough features that make it a viable option for occasional use.



http://remmina.sourceforge.net

Thunderbird

Another gem from the Mozilla Foundation, *Thunderbird* is one of the best email clients, being easy to setup and is brimming with features. Simple setup wizards aid syncing with popular web-based email services and it can manage multiple accounts, supports encryption and is extended through add-ons.

www.mozilla.org/thunderbird

KeepassX

Trying to remember different passwords for the various services is a challenge for most humans (that don't count cards in Las Vegas for fun). You can defer this task to *KeePassX* which stores password in an encrypted database. It can fill in the password automatically and also includes a random password generator.

www.keepassx.org

BleachBit

A distro accumulates a lot of digital gunk over time. *BleachBit* helps you spring clean it and protect your privacy. It also removes temporary and other unnecessary files, and has tools to securely delete files or wipe them.

http://bleachbit.sourceforge.net

OpenSSH

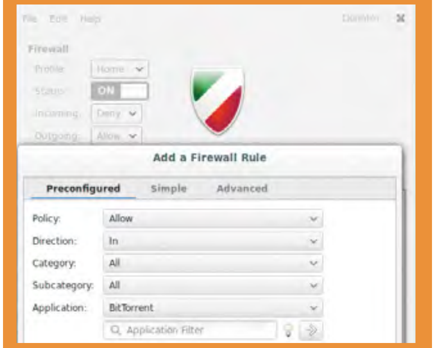
When you need to interface with a remote computer, you cannot do without OpenSSH. It's a family of tools that provides secure tunnelling capabilities by encrypting all traffic and includes several authentication methods, and supports all SSH protocols.

www.openssh.org

Gufw

You may not be using a firewall currently, and if that's because they are difficult to set up then you need *Gufw*. It features an intuitive graphical interface for managing the inbound and outbound traffic rules for various apps and services and even individual ports. Its wizard-like graphical menus are designed especially for inexperienced users.

www.gufw.org



Profile: Home
Status: ON
Incoming: Deny
Outgoing: Allow
Policy: Allow
Direction: In
Category: All
Subcategory: All
Application: BitTorrent
Application Filter

› Gufw has profiles and preconfigured rules to aid inexperienced users.

VirtualBox

When *Wine* doesn't cut it you can use *VirtualBox* to run an entire Windows installation inside a virtual machine. The software is also useful for installing experimental apps that you don't want to deploy on a real computer, and for testing other OSes without exposing it to real hardware.

www.virtualbox.org

Clonezilla

This is a cloning solution that's distributed as a live CD and is popular for doing bare metal backup and restoration of individual PCs. It can also deploy an image to multiple computers in a lab. Clonezilla can work with a large number of popular disks, partitions and filesystem types.

www.clonezilla.org

VLC

Distros ship with a functional video player. But if you need more control, there's no beating *VLC*. It supports virtually every video and audio format out there and includes handy CLI tools for advanced users.

www.videolan.org/vlc

PeaZip

PeaZip is a graphical archiving tool that can work with over 130 different types of archive files and can even create encrypted ones. It integrates with popular desktops and also has a CLI for advanced users.

http://bit.ly/PeaZipSF

Gparted

Use *Gparted* to restructure a disk on your computer. It's available as a live CD and can also be installed inside your distro. *Gparted* can create, resize, move, delete, reformat or check partitions and supports many filesystems.

www.gparted.org

ZuluCrypt

Create an encrypted disk within a file or within a non-system partition or USB disk. *ZuluCrypt* has an intuitive user interface and can be used to encrypt individual files with GPG.

http://bit.ly/zuluCrypt

HomeBank

This is a feature-rich finance app. It can import data from other apps and bank statements in popular formats. It can also detect duplicate transactions and features dynamic reports and is easy to use for budgeting.

http://homebank.free.fr

Internet apps

Get the best of the web with these tools.

Firefox

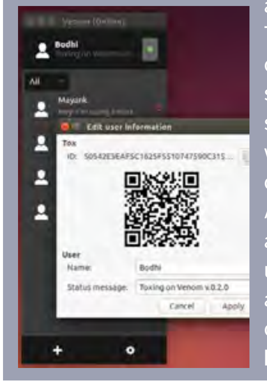
One of the most widely recognised pieces of open source software, Mozilla's *Firefox* web browser is the default browser on virtually every Linux distro. It's pretty responsive and known for its privacy features. You can customise it to the hilt and also extend it with an impressive number of extensions.
www.firefox.com

gFTP

The *gFTP* client is a feature-rich client that'll get the job done, if you need to download files via FTP occasionally. It has a simple two-pane interface that shows the content of the local and remote filesystem. Using *gFTP* you can also transfer files between two remote servers.
http://gftp.seul.org

Tox

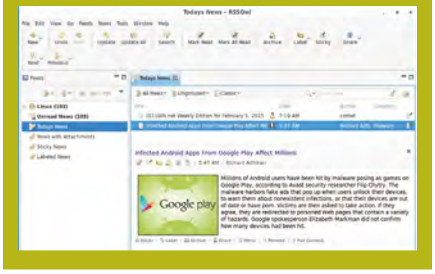
Privacy conscious users should try the new decentralised IM and VoIP client called Tox. This relies on a distributed network, which uses P2P connections, the same technology used by BitTorrent to provide a direct connection, between users for chats and, unlike other Skype alternatives, Tox uses no centralised servers or supernodes, which could be compromised. All chats are also encrypted using the peer-audited NaCl crypto library.
https://tox.im



Tox
ID: 50542E3EAF5C1625F5510747590C315...
User
Name: Bodhi
Status message: Toxing on Venom v.0.2.0

RSSOwl

An excellent desktop alternative to Google Reader, RSSOwl is a news aggregator for RSS and Atom News feeds that's easy to configure. The app gathers, organises, updates, and stores news in an easy to use, and saves selected items for offline viewing and sharing.
www.rssowl.org



Infected Android Apps From Google Play Affect Millions

Millions of Android users have been hit by malware posing as games on Google Play, according to Avast security researcher Filip Chytry. The malware harbors fake ads that pop up when users unlock their devices, to warn them about nonexistent infections, or that their devices are out of date or have porn. Victims are then asked to take action. If they agree, they are redirected to poisoned Web pages that contain a variety of hazards. Google spokesperson Elizabeth Markman did not confirm how many devices had been hit.

Jitsi

Jitsi is the best VoIP app, as long as you're not adverse to Java apps. It supports IM and make one-to-one audio and video calls, as well as audio conference calls. It supports many of the widely used IM and telephony protocols, including SIP, XMPP, AIM, ICQ, MSN, etc. Jitsi has all the features you'd expect from a softphone, and more, such as encrypt text chats with OTR and voice and video by establishing a ZRTP session.
https://jitsi.org

Aria2

What makes *Aria2* a unique utility is that it can download the same file at the same time using different protocols. The lightweight CLI app can download via HTTP, FTP, BitTorrent and Metalink and can also open multiple connections to download the file faster.
http://aria2.sourceforge.net

Midori

The go-to browser for anyone concerned about resource consumption, *Midori* is popular with lightweight distros. Despite its lightweight nature and design, Midori has all the features you'd expect from a web browser including a speed dial, tabbed interface, bookmark management and configurable web search as well as an incognito mode.
www.midori-browser.org

FileZilla

For those who use FTP a lot, there's FileZilla. The client supports FTP, SFTP and FTPS protocols and has just about any configuration option you can imagine. It also has a tabbed interface so you can browse more than one server and even transfer files simultaneously between multiple servers.
https://filezilla-project.org

Deluge

BitTorrent is popular for downloading Linux distros and there are numerous download clients. One of the best is *Deluge* which has multiple front-ends, including a graphical and a web-interface. It has features that enable advanced users to tweak it to their liking and also has a nice library of plugins.
www.deluge-torrent.org

Pidgin

Pidgin is a wonderful app for instant messaging over many network protocols. You can sign in with multiple accounts in the single client and chat with many friends in different networks. You can use it to connect to AIM, MSN, Google Talk, Yahoo, Jabber, IRC and more chat networks all at once.
www.pidgin.im

Games

0 A.D.

This is a real-time civilisation-building strategy game that features impressive graphics and intense battle gameplay. It's yet to have a final release but has already won accolades in its current state.
http://play0ad.com

FreeCiv

Another strategy game that challenges players to lead their tribe 4,000B.C to the space age.
www.freeciv.org

Alien Arena

A popular first person shooter with a sci-fi theme and the tournament style deathmatch of *Quake* and *Unreal Tournament*. The game has several game modes and over 60 maps, and is quite configurable.
http://red.planetarena.org

OpenMW

OpenMW is a new game engine that recreates the popular *Morrowind* RPG. The aim of the project isn't to improve game assets or add additional features but to provide gamers a more moddable edition of the game.
https://openmw.org

FlightGear

For fans of aircraft simulators there's *FlightGear* that aims to offer flight across real world terrain. It includes scenery for more than 20,000 airports, and can be extended with your own aircraft and locations.
www.flightgear.org

Office and productivity

Enhance your workflow with these apps.

Calligra

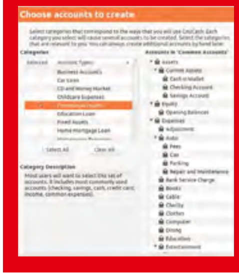
Unless you feel you need *LibreOffice*'s superior compatibility with proprietary formats, you may want to consider *Calligra*. It's a continuation of *KOffice* and unlike *LibreOffice*, *Calligra* has a modern-looking, modular design, and also uses Open Document as its native file format. It ships with a large clutch of apps. In addition to the *Words* word processor, *Tables* for spreadsheets, *Stage* for preparing presentations, and *Kexi* for managing databases, it also benefits from Krita [*see p40*] for digital painting.

www.calligra.org

GnuCash

Gnome users have *GnuCash* which is similar to *KMyMoney* in terms of features, but also handles and categorises entries differently. *GnuCash* is a personal and small business accounting app that's based on double-entry for professional reporting and besides dealing with monetary transactions, it can track things such as stocks, bonds and mutual funds.

www.gnucash.org



Select categories that correspond to the ways that you will use GnuCash. Each category you select will cause several accounts to be created. Select the categories that are relevant to you. You can always create additional accounts by hand later.

Most users will want to select this set of accounts. It includes most commonly used accounts (checking, savings, cash, credit card, income, common expenses).

Xournal

This app is very handy for when you need to scribble bits of information down for later. As well as typing out notes, you can use it with either a mouse or a stylus. It can also be used to add annotations to PDF files.

http://xournal.sourceforge.net

Zathura

This is a simple and a lightweight PDF reader that supports almost all the usual features you'd expect. You can search text strings, jump pages, zoom in and out, rotate pages, add bookmarks and more. In addition to PDFs, it can display DjVu and even encrypted documents.

https://pwmt.org/projects/zathura

AbiWord

The wide gap between rich text editors and word processors is occupied by *AbiWord*. It's lightweight but still offers commonly-used word processing features, which makes it a popular for lightweight distros. It also offers cloud-based collaboration capabilities via its **AbiCollab.net** service.

ProjectLibre

A project management tool helps you stay on top of ongoing projects and *ProjectLibre* is one of the best. It's an award winning app that's used widely by many enterprises around the world. *ProjectLibre* has several useful features and can also visualise tasks with various charts and reports.

www.projectlibre.org

OpenLDAP

OpenLDAP is great for when you want to run a directory server. It implements the LDAP protocol and has all the expected features, including logging, replication, access control, user and group management etc. It also integrates with Active Directory.

www.openldap.org

Gnumeric

A *biWord* is usually paired with the lightweight Gnumeric spreadsheet app. However, the app isn't light on features and offers a lot more functionality than proprietary spreadsheet apps. *Gnumeric* will import data from Microsoft Excel files and there are import filters for other apps as well.

www.gnumeric.org

KMyMoney

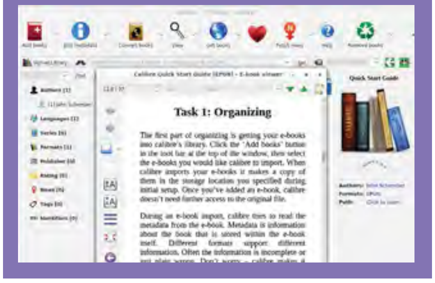
Designed for KDE users, *KMyMoney* is a feature-rich accounting app. It supports different account types, such as Cash, Checking, Savings, etc and can categorise incomes and expenses, and can reconcile bank accounts. If your bank allows it, you can have *KMyMoney* connect to your bank directly to retrieve your account activity.

https://kmymoney.org

Calibre

You can use *Calibre* to manage your collection of ebooks, and supports a wide range of readers and smartphones. The app can import ebooks manually or, if you prefer, by syncing a reading device such as the Kindle. Any files imported can be sorted and grouped by metadata fields, which can be pulled from various online sources, such as **www.goodreads.com**.

www.calibre-ebook.com



Task 1: Organizing

The first part of organizing is getting your e-books into calibre's library. Click the 'Add books' button in the tool bar at the top of the window, then select the e-books you would like calibre to import. When calibre imports your e-books it makes a copy of them in the storage location you specified during initial setup. Once you've added an e-book, calibre doesn't need further access to the original file.

During an e-book import, calibre tries to read the metadata from the e-book. Metadata is information about the book that is stored within the e-book itself. Different formats support different information. Often the information is incomplete or

Achievo

This is a web-based resource management tool with a simple interface for accessing its CRM, HRM and project management and planning tools. You can also track resources across multiple projects.

www.achievo.org

Okular

The default PDF viewer for KDE and includes a good number of useful features. Besides PDF it can also read a number of other file types, including Postscript, DjVu, CHM, XPS, ePub, TIFF, CBR, and others.

https://okular.kde.org

LaTex

LaTex is a document preparation system and document markup language based on TeX. Its purpose is to simplify TeX typesetting for documents containing mathematical formulae and is widely used in academia.

www.latex-project.org

Shutter

Besides capturing the full screen, Shutter can capture a specific area, or a window. You can also upload to a hosting service.

www.shutter-project.org

ClamAV

While most viruses and trojans will have no effect on Linux, you still can have infected files in your distro that can wreck havoc when accessed on a Windows machine. So be a good admin and use ClamAV to scan files.

www.clamav.net

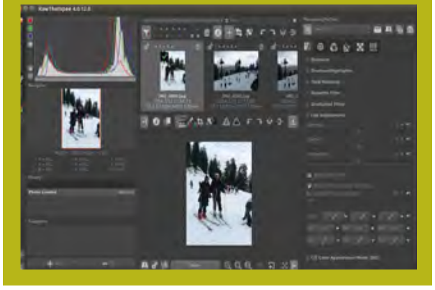
Hobbyist

Follow your passion.

RawTherapee

Do you shoot with a digital single lens reflex camera (DSLR)? Then take a look at *RawTherapee* which includes a wide range of tools for processing and converting RAW files. In addition to basic manipulations, the app has extensive options for working with RAW files. Using the app you can adjust the colour and brightness values of your images, correct white balance, adjust tones, and a lot more. Besides RAW files you can also use *RawTherapee* for editing traditional image files, and it also includes Adobe Lens Correction profiles.

www.rawtherapee.com



Scribus

A comprehensive desktop publishing program. *Scribus* can be used to create professional press-ready online and print documents including brochures, booklets, books and magazines. It has a feature-rich interface and has features, such as PostScript colour separations, support for CMYK and spot colours, ICC profiles, and printer marks. *Scribus* also includes a variety of templates and styles and you also get an array of settings and tools to precisely define and position the various layout elements you require.

www.scribus.net

Krita

Although *Krita* is part of the *Calligra* suite (*see p39*), it needs a special mention of its own. *Krita* is a digital painting and illustration app that offers many expressive brushes, HDR painting, filters, perspective grids, painting assistants, and many other features you'd expect from such an app.

www.krita.org

Stellarium

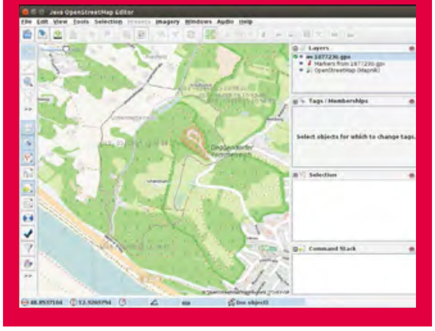
Stellarium is a free open source planetarium for your computer. It calculates the positions of the Sun and Moon, planets and stars, and draws the sky as per the users location and time. It can also draw the constellations and simulate astronomical phenomena such as meteor showers, and eclipses.

www.stellarium.org

JOSM

Keen to contribute to the mapping project, OpenStreetMap? Then use *JOSM*. It's a Java-based offline map editor that can help you plot GPS traces. You can load GPS track-logs into JOSM and start adding streets to OpenStreetMap instantly. Although OpenStreetMap has several other editors available, most contributors use JOSM for their edits, as it lets them upload changes back to OSM quickly and easily enough. JOSM offers several features and can be extended with plugins and styles.

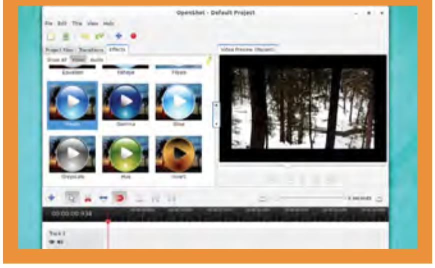
https://josm.openstreetmap.de



OpenShot

There are only a handful of video editors for Linux and *OpenShot* offers the best combination of features and ease of use for the home user. You can use it to combine videos, audio tracks, and still images together and add in captions, transitions, and more, and export the final product in a variety of formats. *Openshot* can also use *Blender* to create 3D animated titles

www.openshot.org



Inkscape

Another pro-quality tool, *Inkscape* offers advanced vector graphics editing and is popular for drawing vector art, line art, and designing logos and graphics. It's brimming with features, such as markers, clones, alpha blending and more, and is often compared to expensive proprietary apps such as *Illustrator* and *CorelDraw*.

www.inkscape.org

Cinelerra

Cinelerra is excellent if you need to edit more than home videos, as it's the most advanced non-linear video editor and compositor for Linux. It supports HiFi video and audio and is resolution and frame-rate independent, which enables it to edit videos of any size. The app has several advanced features, such as overlays, denoising, normalisation, time stretching, color balance, compositing, real time effects and a lot more. It also includes a compositing engine for performing tasks such as keying.

www.cinelerra.org

Media

Comix

Digital comics are distributed as comic book archive files that mainly consist of a series of image files, typically PNG or JPEG files, stored as a single archive file. *Comix* can read digital comics in virtually every format.

http://bit.ly/ComixApp

FontForge

FontForge is a feature-rich app for creating and editing fonts and supports all common font formats. It can extract information from a font file as well as convert from one format to another, and can be used for previews.

http://bit.ly/FontForge

CairoDock

CairoDock is a MacOS X dock-like app. One of its main advantages over other docks is that it doesn't require a compositing window manager to work and can add bling to older low-powered machines.

www.glx-dock.org

Audacity

If you need to work with audio, you should use the powerful A*udacity* sound editor. You can trim audio, combine tracks, and even stack multiple tracks, as well as export to a number of formats and quality settings.

http://bit.ly/AudacityApp

MPD

The Music Player Daemon is an audio player with a server-client architecture, which means you can control it remotely from another computer. It plays audio files, organises playlists and can maintain a music database.

www.musicpd.org

Development

Power tools and programs for power users.

jEdit

This is a text editor for programmers that supports auto indent, and syntax highlighting for more than 140 different programming languages. The app enables you to define complex macros and offers a powerful and user-friendly keyboard mapping system. It's highly configurable and customisable, and you can extend its functionality by adding plugins.

www.jedit.org

Eclipse

There's no beating *Eclipse*, the most feature-rich IDE. Although Java is its speciality, Eclipse supports a range of languages via plugins. In fact, its plugin marketplace is an indispensable resource. Eclipse does code refactoring and you can use it to extract the selection as a local variable or method. Since it can target multi-person installs, it handles version control very maturely

www.eclipse.org

BlueFish

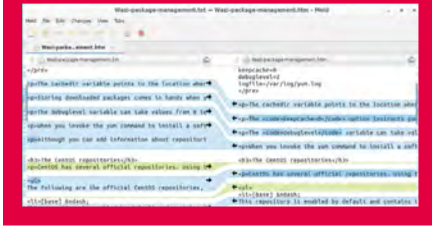
Do you develop for the web? *Bluefish* is a multi-language editor that's designed for web developers. It supports many programming and markup languages and focuses on dynamic and interactive websites. It supports code block folding, unlimited undo/redo, automatic tag closing, and syntax highlighting. Another useful feature is the snippets bar from where you can add the most common snippets of code for a variety of languages. *Bluefish* also has support for popular open source web apps such as MediaWiki and Wordpress.

http://bluefish.openoffice.nl

Meld

A graphical diff tool, *Meld* enables you to compare two or three files as well as whole directories. It includes features, such as syntax highlighting and direct file editing, and using the tool you can easily isolate and merge the differences. *Meld* can also be used to browse various popular version control systems such as *CVS* and *Subversion*.

www.meldmerge.org



Wazi-package-management.txt — Wazi-package-management.htm - Meld
Meld File Edit Changes View Tabs

```
</pre>
<p>The cachedir variable points to the location wher
<p>Storing downloaded packages comes in handy when y
<p>The debuglevel variable can take values from 0 to
<p>When you invoke the yum command to install a soft
<p>Although you can add information about repositori
<h3>The CentOS repositories</h3>
<p>CentOS has several official repositories. Using t
<ul>
The following are the official CentOS repositories.
<li>[base] &ndash;
```

```
keepcache=0
debuglevel=2
logfile=/var/log/yum.log
</pre>
<p>The cachedir variable points to the location wher
<p>When you invoke the yum command to install a soft
<h3>The CentOS repositories</h3>
<ul>
<li>[base] &ndash;
This repository is enabled by default and contains t
```

KompoZer

New and experienced HTML programmers will save a lot of amount of time and effort with the *KompoZer* editor. It has an intuitive interface and includes a colour picker, an FTP site manager, CSS editor, customisable toolbars, forms, spell checker, markup cleaner and can also validate code using W3C's HTML validator.

www.kompozer.net

Gimp

Despite its name, Gimp is a powerful, comprehensive image manipulation program. It offers a wide range of tools for professional-quality photo retouching and image manipulation capabilities for free. It also offers a huge list of features and supports all the common graphics file formats.

www.gimp.org

Blender

With *Blender* animators can create 3D printed models, visual effects, art, interactive 3D applications and video games. The app provides a wide range of features that can be used to create 3D animation films. It's a one-stop 3D package and includes a gaming engine, a video sequence editor, production-ready camera and object tracking, a large library of extensions, and an advanced physics engine. It can render fluid dynamics and simulate the movement of elastic objects and clothes.

www.blender.org

Geany

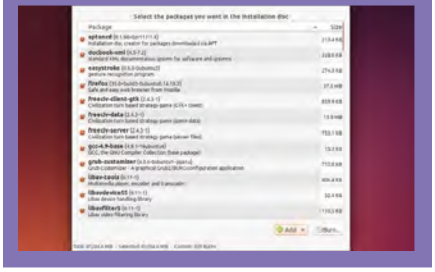
You don't need a full-blown IDE if you only program occasionally, which makes *Geany* a good choice. It's a cross between a plain text editor and an IDE with support for the popular languages and nifty features like a compile/run button, a listing of functions defined in the currently opened file, and much more.

www.geany.org

APTonCD

Suddenly realise that you need to move your Ubuntu installation or need to give a friend a copy of your setup? With *APTonCD* Ubuntu users can back up all of their installed packages to an ISO image, which can then be added as a software source on another installation. You can use this source to restore the packages on to the system or keep everything in the APT cache.

aptoncd.sourceforge.net



Clementine

Use Clementine to play locally stored music and streaming audio. The app has an attractive interface and it also helps organise and transfer music to various devices, and integrates well with popular cloud services.

www.clementine-player.org

Icecast

With *Icecast* you can stream music across the network. *Icecast* supports many audio streams simultaneously and listeners can access a stream via a remote media player and also configure MPD as a source.

www.icecast.org

Amarok

If you use KDE your distro may already include this music player, Amarok. It too integrates with several online audio services, and its features include creating dynamic playlists, bookmarks, scripting, context view.

https://amarok.kde.org

LMMS

LMMS is digital audio workstation that produces music by synthesising sounds, arranging samples, and playing them on a MIDI keyboard. It also has a song editor and plugins to simulate instruments and effects.

www.lmms.io

Kodi

Until recently *Kodi* was known as *XBMC*. It's an excellent option for users who wish to turn their PCs into media hubs. It plays most kinds of media files and works with TVs, IR and bluetooth remote controls.

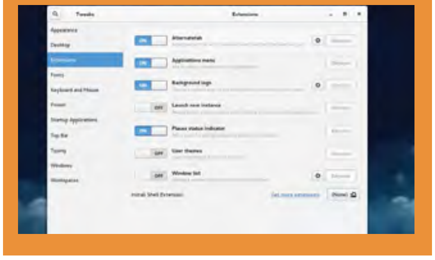
www.kodi.tv

Utilities

Apps that let you do more with your computer.

Gnome Tweak Tool

Not satisfied with the stock Gnome desktop? Use the *Gnome Tweak Tool* to customise several aspects, including the appearance settings of the desktop. With this tweak app you can also change the behaviour of the Windows and Workspaces, manage extensions and you can even circumvent the design philosophy of Gnome 3 by placing icons, files and folders on the Gnome desktop. **http://bit.ly/GnomeTweakTool**



digiKam

One of the best photo management tools for Linux is *digiKam* and it has features that'll appeal to all kinds of users. It recognises all major image file formats and can organise and sort images based on metadata. The app also has plugins to export images to various online services. **www.digikam.org**

K3b

Although it's designed for KDE, the *K3b* optical media burning utility is one of the finest for the job. The app can burn multiple El Torito boot images, audio CDs, VCDs, SVCDs, mixed-mode CDs, eMovix CDs, and DVDs. It can also rip DVDs and write ISO images. **www.k3b.org**

Grub Customizer

Grub 2 is the most popular Linux bootloader that's used by virtually all major distributions. It's an impressive piece of software with lots of options. The *Grub Customizer* is a simple to use graphical tool,which enables you to quickly customise all aspects the bootloader, including its appearance. **www.launchpad.net/grub-customizer**

DOSBox

Relive the good ol' days with *DOSBox* and play your favourite classic DOS games that won't run on your modern hardware. This is an x86 PC emulator that creates an IBM PC compatible computer complete with compatible graphics and sound cards. The app can also simulate networking hardware for multiplayer games on the local network and even over the Internet. The *Wine* project even uses code from *DOSBox* to bolster support for DOS apps. **www.dosbox.com**

Avidemux

Avidemux is a video editor and converter that can be used for basic cutting, filtering and encoding tasks. It supports many file types, including AVI, MPEG, and MP4. The app is designed for users who know what they want to do but also provides an intuitive interface so that tasks such as cutting and appending videos are pretty straightforward. The app has some presets and users can also save custom settings that make the app easier for new users to operate. **http://fixounet.free.fr/avidemux**

Handbrake

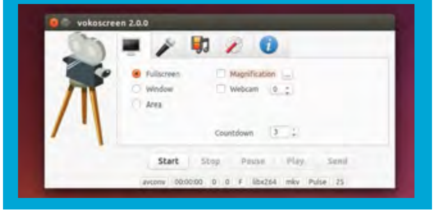
When the need to convert a video arises, *Handbrake*, the video transcoder app does a commendable job. It can convert nearly any format and supports a wide range of video codecs. One of its best features is built-in device profiles for popular devices that make the conversion process easier. **www.handbrake.fr**

EasyStroke

Want to control your PC with the flick of the mouse? The *EasyStroke* app lets you define and manage gestures by recording the movements of your pointing device while holding down a specific mouse button. You can then configure actions that'll be executed when the app recognises the defined stroke. **https://easystroke.sourceforge.net**

Vokoscreen

A feature-rich screencasting app worthy of note is *Vokoscreen*, which is based on FFmpeg for handling multimedia data. *Vokoscreen* can capture both video and audio, with options to record the entire screen, window or a selected region, along with video from a webcam. The app supports MPEG4, x264, MP3 and Vorbis codecs and can save files in either .AVI and .MKV containers. The app offers some controls such as the ability to change the video quality and frames captured per second and can be used to make screencasts of games. **www.kohaupt-online.de/hp**



Terminal

Ncmpcpp

This is a command-line *MPD* client that's easy to use and customisable. It provides useful features such as the ability to sort playlists, song lyrics, item filtering, fetching artist's info from last.fm, tag editor and much more. **http://bit.ly/Ncmpcpp**

Samba

Samba is a suite of programs that enables Linux users to access and use files, printers and other commonly shared resources on a Windows PC on a network and does this by supporting the SMB protocol which. **www.samba.org**

rTorrent

Here we have a command-line BitTorrent client with an ncurses interface. You can run it as a daemon and manage it with *screen* and since it supports SSH you can manage your torrents from any remote machine. **http://bit.ly/rTorrent**

Links2

There are lightweight browsers and then there's *Links2*. This is a web browser that can render complex pages and even has a pull-down menu. It's also special because it's a CLI browser that you operate via the keyboard. **http://links.twibright.com**

Midnight Commander

Before the days of graphical file managers, real hackers used *Midnight Commander*, known as *mc*. It's still your best option if you regularly find yourself in the console environmen a lot. **http://bit.ly/MidnightCdr**

Admin tools

Take charge of your distro with these power apps.

Redo Backup

We've mentioned the *Clonezilla* cloning solution earlier in the feature [*see p37*], but if all you need is a tool to swap out an old disk for a new one, then you use *Redo Backup and Recovery*. The tool is designed for inexperienced users and has the simplest of interfaces.

www.redobackup.org

XAMPP

The XAMPP stack gives you a single package that you can use as a sandbox to test and develop web apps. It includes all the necessary components such as *Apache*, *MySQL*, PHP, and Perl as well as several other libraries, modules and tools, such as *phpMyAdmin* and *FileZilla* for managing the stack components. Once installed, you can manage the various services via a graphical control panel.

www.apachefriends.org

Déjà Dup

The app's minimal GUI sets itself apart from the various other backup apps you'll find, and it lets you configure backups within a matter of minutes. *Déjà Dup* is based on *Duplicity* and provides just the right number of features for desktop users who aren't used to the ways of a backup tool.

http://live.gnome.org/DejaDup

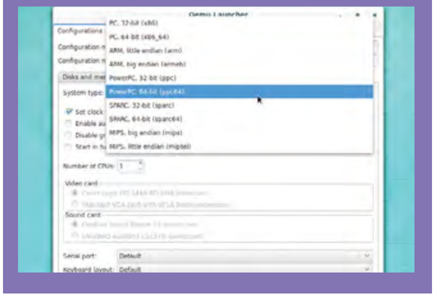
No recent backups.

Next backup is today.
A backup automatically starts every week.

Qemu

It's is a feature-rich multi-purpose processor emulator and virtualiser. You can use it to create virtual machines and even emulate various hardware architectures. If you have the right hardware on tap (a processor with hardware virtualisation extensions), you can use *Qemu* with *KVM* in order to run virtual machines at near-native speed.

www.qemu.org



Mondo Rescue

Mondo is a unique backup solution that creates bootable backup and restoration disks customised for the system being used. *Mondo* has a text-driven interface and works with a wide range of file systems and can use a variety of media as backup mediums.

www.mondorescue.org

Open Media Vault

When you need more protection for your data than a simple backup then you need to deploy a NAS server. The *Open Media Vault* project is a Debian-based server that offers the power of commercial options in a way that's easy to setup and manage.

www.openmediavault.org

Conky

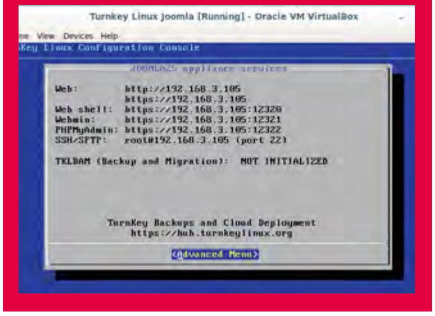
Concerned about the resource utilisation on your PC? *Conky* is a nifty little app that lets you keep an eye on your system. It can monitor and report on the states of various components. The tool is very flexible and highly configurable and can also display information from apps, such as weather updates.

http://conky.sourceforge.net

Turnkey Linux

The Turnkey project produces appliances which you can use to deploy a new server in a jiffy. A Turnkey appliance is a self-contained system that packs in a fully functional web app that runs on top of Just enough Operating System (JeOS) components required to power that particular app. All the appliances are based on Debian but are available in several formats depending on the hardware that you want to deploy it on. Once they're up and running you can manage each appliance using a browser-based interface.

www.turnkeylinux.org



```
Key Linux Configuration Console

              JOOMLA25 appliance services

Web:          http://192.168.3.105
              https://192.168.3.105
Web shell:    https://192.168.3.105:12320
Webmin:       https://192.168.3.105:12321
PHPMyAdmin:   https://192.168.3.105:12322
SSH/SFTP:     root@192.168.3.105 (port 22)

TKLBAM (Backup and Migration):  NOT INITIALIZED


          TurnKey Backups and Cloud Deployment
              https://hub.turnkeylinux.org

                    <Advanced Menu>
```

Zentyal

The Zentyal distro has all the components you need to run a gateway server. The distro simplifies the process of setting up, monitoring and controlling the components of the server with a host of custom management tools and helps you configure the servers without mucking about with config files.

www.zentyal.org

Mutt

Mutt is to email what *Links2* is to the web browser. It's a text-based mail client that is highly configurable and it supports both POP and IMAP protocols and has all the usual features you'd want from an email client.

www.mutt.org

Profanity

Profanity is a console-based client for the XMPP protocol that supports multi-user chats and OTR message encryption.

www.profanity.im

Canto

Want to do more from the command-line? Get the *Canto* CLI RSS feed reader. It supports RSS, Atom and RDF feeds and imports and exports feeds in OPML format. It has lots of customisation and even configure it with Python.

http://bit.ly/CantoRSS

mpg123

This is an MP3 audio player for the command-line that supports gapless playback. It's so good that its decoding library, libmpg123 is used by other audio players for MP3 playback

www.mpg123.de

FFmeg

One of the most versatile media conversion utilities, *FFmeg* can manipulate virtually any type of media file in various ways, such as changing bitrate, extract audio, record streams, extract stream and much more.

www.ffmpeg.org

National Service

Sam Tuke talks budget band aids and health record healing with the National Health Service's own open source crusader **Peter Coates.**

LINUX FORMAT Interview

Peter Coates has spent a year convincing Britain's largest employer to buy open source and its largest suppliers to sell the option. As Head of the NHS Open Source Programme, he's fostered a generation of new in-house apps, and witnessed some of the largest companies in the field climb on board with copyleft. His work has cleared a path for wider adoption, with a new toolkit for public sector buyers, and a novel support structure for community made apps. And with a 20-year record in public sector IT he's got stories to tell.

Linux Format: **What's your background?**
Peter Coates: I was a Head of IT and Information for many years at a large local authority in the social services department. I had complete control over the IT side of things, from boxes and wires, telephony, all the way through to desktops, managing information systems, and training. I came to realise that the public sector is not well served by software. It's a very, very small market. English social services departments work quite differently to other ones, even in the UK, never mind outside of Europe, because they are joined up with all sorts of other government services.

Therefore the market for companies who supply software is also very small. That means there isn't much innovation, because there's no significant investment in research and development – there isn't a good enough prospect of recovering that investment. All this means that, as an IT manager, to meet the needs of your organisation you're looking at a lot of bespoke development.

ON DEVELOPMENT
> “We would have had to write something brand new from scratch, with no support...”

So local authority, the public sector, they'd try using well-known systems for business administration. Local authorities would use SAP and Oracle Enterprise Resource Planning applications (ERPs), generally wall-to-wall Microsoft for office productivity, email etc. They could get away with that approach, though they weren't ideal, and were very expensive. But they'd really struggle with the clinical space – systems that deal with the cases of individual patients and related services.

The only option they were left with was to use a small number of system vendors who catered to that market. These were also expensive for the budgets they had. It meant we actually had to develop things ourselves – additional modules to meet our needs, for example. Then we started coming up against issues, commercial issues, around being able to integrate additional functionality with the core systems we got from software vendors.

At that time APIs weren't commonly available, which would have allowed us to connect different proprietary components more easily. Where such interfaces did exist, they were expensive and not very open. My social workers had very particular needs. Proprietary solutions weren't available to buy; we had to develop something. And we had to be able to do it rapidly.

About 20 years ago I put together a development team to add custom functionality. Open source wasn't yet on my radar. Whatever we made had to connect to the other core systems that were already in place. And that's where we'd start to hit roadblocks due to licensing restrictions. Over the following years we worked with these constraints. Some of the applications we created were quite successful, but the restrictions didn't go away.

LXF: **How did you get into open source?**
PC: The team was crystal clear from the start that everything we developed would be freely available for other departments. So our work effectively was open source in how we handled it, but it wasn't open source licensed. I was working on a very challenging project to implement the new 101 non-emergency number in Northumberland Tyne and Wear. It had to connect securely to a whole host of local systems, including Police Command and Control systems. In the end we found an open source solution called ActiveMQ. This was the first software I'd used that I could take, consider, implement, and share. It was a revelation! We could just take this free software that had a large community, language support, development tools, and adapt it to our own very, very specific needs.

LXF: **And did it work?**
PC: Actually, it amazed me that it had already reached such a mature level – it was very stable. If it hadn't been for this open source app, we would have had to write something brand new from scratch, with no support, and no community. Making up a custom solution like that would have been extremely expensive.

At that point, open source became a third way for me. An alternative to having to either develop from scratch, or buy ready made. The system we built ended up running on a Linux-*Apache* infrastructure, because that turned out to have far superior performance than the other systems we tested. These two things gave me a window into how valuable open source could be for us.

LXF: **What does the NHS Open Source programme do?**
PC: Primarily we're an educational service for informing people of the benefits that open source can bring. We provide resources, expertise and advice to healthcare professionals. We exert influence over the entire healthcare community around the use of open source. Ultimately this education is a means to an end; the goal is to help deliver better healthcare services and outcomes for patients. »

» Two policies drive the programme, both from a department called the Government Digital Service (GDS). They have a national strategy to ensure that open source has a level playing field when competing against proprietary systems. That's the first one. The other is about maximum benefit from public spending. If tax payer's money is going to be spent building software, then that software should be open source, so that there is maximum opportunity for re-use. Of course, there are some areas of government where that's just not appropriate.

We're trying to do things differently to how they're being done in the NHS today, because if you do what you've always done, you get what you've always got. This is about transformation to a new model.

LXF: Who's your boss?
PC: The programme answers directly to NHS England, which is responsible for organising and paying for all services in that region.

LXF: You joined the programme 6 months after its creation, and have just completed your first year at the helm. What are you most proud of?
PC: One thing is the new Commercial Toolkit, created in-house, for obtaining open source for health organisations. It explains how to implement and maintain open source solutions, and how to procure deployment and support services. This is critical information for NHS organisations that want to switch.

Another thing we're proud of is a delivery model we've developed for applications that come from a community rather than from an individual company. This is based on not-for-profit organisations acting as custodians for open source apps. They provide assurance to healthcare users that the software is safe to use, even though there isn't a company that's behind it.

LXF: Why is such a kit necessary?
PC: There are perception issues and myths surrounding open source which we need to bust through. Myths like "Open source software is not as safe as proprietary software", and "procuring open source is harder". The NHS is a huge and diverse landscape, and open source is still new and unknown to large parts of the public sector.

Misunderstandings like this exist on the supply side too – it's not just the buyers. We provide information to proprietary software vendors and put them in touch with open source vendors and integrators, to show them how other companies have moved from being product-based to being service-based. Companies come to our events to see if there's a new commercial model that would suit them.

You've got to remember that for a long time in the NHS, software choices were limited to what was effectively a catalogue of proprietary systems. We're coming out of that period now, and organisations have got a lot more choice. The kit also helps navigate compliance with things like procurement regulations. It's a comprehensive document.

ON MEDICAL STAFF CONTRIBUTIONS

> **"They're passionate about making systems that meet their patients' needs."**

LXF: What's the not-for-profit assurance model that you mentioned?
PC: It's a way to provide assurance for open source software that would otherwise be wanting. It uses a particular legal structure, based on Community Interest Companies (CICs) – not-for-profit organisations, to fulfil the role of custodian for a particular application that's used in healthcare. This way people who want to use that software have someone to go to for assurance and services related to the app. Questions about future releases, testing, and so on can be answered by the CIC. They work as umbrella organisations to represent the needs of users in healthcare and communicate with the software's developers.

LXF: How many Community Interest Companies are there?
PC: Five are being set up currently. One example is the Open Eyes Foundation, which already has multiple sponsors. Another was set up for openMaxims, a large Electronic Health Record (EHR) system released under the AGPL last June. Three NHS Trusts, including Blackpool Teaching Hospitals, became legal members in February. More are in the works for software used in dental and mental health, and community dentistry.

LXF: How do you get proprietary software companies involved in what you're doing?
PC: We run events with trade bodies, such as TechUK, which represents 850 technology companies. We've held a number of open days and open sessions for vendors, where we talk about open source and the NHS. We present our work at broader conferences like EHI Live, and Health Insights. I'd say we've participated in over 20 formal events since I started. Additionally, we have our own NHS Open Source Open Days, two last year at Chelsea FC and Newcastle Racecourse.

LXF: Are you involved with other open source communities?
PC: Yes, we went to the MongoDB Public Sector Hackathon in London. Other Central Government departments who were at that event were blown away by what we've been doing. We've also got a seat on the British Computer Society's Open Source Specialist's Group.

LXF: We notice there's an official @NHSOpenSource Twitter account – who're you trying to reach?
PC: I think our Twitter followers fall into two camps: those interested in health informatics, including doctors, nurses etc, and then vendors and suppliers, including many proprietary ones. Somewhere in there is a group of people who develop health apps in their spare time. It's very easy to get engineers interested in working on new things. Some of them do it for their own technical interest, with the blessing, if you like, of their employer. Many of them work for the large proprietary vendors. When they attend our events it's voluntary. I think it's like a hobby for them.

LXF: Do you prefer a particular licence for community contributions?
PC: The AGPL (Affero General Public License) is one of our preferred licenses because all of the changes must be shared back to the community.

LXF: What's been hardest thing about raising awareness?
PC: It's a complex message to communicate. Software is not 'one size fits all'. Assurance is key because the organisations who're using this stuff do very serious work: every day the lives of patients depend on them. They need to be confident that anything they use, whether it's proprietary or open source, is safe. They need assurance during the procurement process that there are sufficient professional organisations

around to provide support, to implement it, to extend it. That's been the biggest challenge.

What's been surprisingly easy is getting clinicians – hands-on staff like doctors – to contribute. They're really passionate about software. They're passionate about making systems that meet their patients' needs, and they're prepared to give their time. They are really the driving force. We're just here as a facilitator and enabler.

LXF: If assurance is key, are enough companies offering those services for open source?

PC: The trouble is that most of the organisations who support open source in England have not traditionally worked in the health sector. We definitely need the availability of more open source solutions, and more vendors who can provide open source services to the health sector. In other parts of the world this is already happening. VistA is an open source EHR system used and developed by the US Department of Veterans Affairs. It delivers healthcare to all ex-US service personnel – it's a very large system. In North America, it's supported by the big IT vendors, including PricewaterhouseCoopers, HP and Google.

LXF: How do you encourage vendors to switch to open source products?

PC: I explain that their products would be more attractive to customers if they were free from the threat of vendor lock-in. Businesses developing open source products are effectively saying: "we're so confident that you'll want our services, we'll let you choose where you buy – our offer is that good". It's a compelling pitch.

Some vendors try dipping their toe into the open source waters with a new product, as an experiment. New products don't have a user base, and the company wants to bring it to market. They say "Actually, I'll release it as open source, and generate business revenue from value added services such as support, implementation and maintenance".

Vendors can also offer proprietary-style contracts for open source products, just as they do for their other products. IMS effectively offers both. They say: "Here's our product openMaxims, you can get a copy from us with a cast-iron guarantee, support and maintenance, but you can have the source code, too". Other vendors make opportunities by adding open source components together and setting up as an implementer or support partner of a third-party app. Because of the licence, they can combine different systems together: integrate their unique product, and reap the benefits of a broader service offer.

LXF: How is the Nation Health Service benefiting from open source?

PC: In my experience clinicians are more likely to contribute to an open source project than they are to a proprietary application. They can choose to participate with the knowledge that their work is not going to be commercially exploited for someone else's advantage. Because the resulting product will be available for everyone's use and adaptation, it'll instead be used to deliver better software, and better healthcare.

I feel this is unique to the medical profession, and why open source is such a good fit for healthcare. Medics are used to sharing their discoveries and practices with a wider community in order to improve them. That's how they work. They take an iterative approach, publishing ideas in a journal, letting others test and improve it, and then publishing new findings in turn. That's how we want our software development to work too: a product is developed, contributed to by others, and ultimately improved upon. An open source licence means commercial exploitation can't prevent that evolutionary process. The goals are different; rather than a purely commercial outcome, it's about the best solution that can be achieved.

LXF: So it's not about the money?

PC: We're focused on the long-term benefits, not short term cost savings on licenses. By taking the open source approach we benefit from all these ideas, collaborations and innovations. When combined to serve healthcare needs we end up with a more useful system that delivers better patient outcomes.

The real financial saving comes from preventative services. When we're able to predict the need for a low-cost intervention early on, and avoid the need for a later high-cost intervention, that's a huge advantage in terms of both the patient's health, and public spending. This is where interoperability issues come into play.

LXF: Incompatible health records have been an issue – can open source fix the problem?

PC: It's a major problem when a proprietary licence impedes a healthcare system's function. Open source gives you the ultimate flexibility here. It's like back when I was an IT Manager – the problem wasn't a technical one, it was a legal one, due to terms and conditions and licenses. The same problem has been recognised by the United States Congress: strict proprietary licensing can impede the healthcare economy.

LXF: Earlier you mentioned Open Eyes. Is that a port of xeyes?

PC: No, it's a web-based Electronic Patient Records system for ophthalmology. It was created by Bill Aylward at Moorfields Eye Hospital. It set out to be an open source project from the start, including it's full web stack. It's now being deployed elsewhere, and has the support of The Queen Elizabeth Diamond Jubilee Trust. Their foundation uses our CIC model, and so far they're doing very well.

LXF: Do you think big vendors will open source more of their products in future?

PC: It's a big decision for a business to open source their property and write off revenue from a licence-based business model. It obviously stacks up for IMS [Health, a US healthcare technology company]. They've just signed a ten-year deal for openMaxims with Ramsay Health Care private hospitals, serving 1,500 users per day.

LXF: Where can I learn more?

PC: There are lots of videos online about what we're up to, and our Twitter feed has carried the latest news, including patient workshops and Open Day announcements. You'll also find us at the next round of Health Insights events.

For more details see:

https://twitter.com/nhsopensource

http://www.technologystrategy.england.nhs.uk/pg/groups/99205

http://www.healthinsights.co.uk. LXF

Run a Linux Business

Keith Edmunds shows how he powers his computing business with open source and how you could too.

Technology has always fascinated me. As a child and teenager it was electronics, and when the chance came to work for Digital Equipment Corporation, a (now defunct) American computing company in 1980 (yes, I am that old), I jumped at it. And I loved it. I learned a lot about computing, even before personal computing really took hold.

When Linux arrived on the scene in the early 1990s, I thought it was brilliant (and I still do). As my career progressed, I left Digital and one day I found myself managing the European IT helpdesk for a worldwide company. We needed to monitor our network to understand when things were failing. I wondered if maybe Linux could help and so yes, I set up a Linux system to do just that. Linux clearly had a value to businesses, and it was around that time (late 1990s) that I began to wonder whether it would be viable to run a company that helped other businesses use Linux? That may not seem like a very revolutionary idea now, but back then Linux was virtually unheard of outside of the IT industry (and for large parts inside it, too). I spent a lot of time thinking and talking about it, and – to cut a long story short – in 2002 I left the job I then had and started Tiger Computing.

> "I wondered: would it be viable to run a company that helped other businesses use Linux?"

At Tiger Computing, we support Linux for businesses throughout the UK, and our own business runs (almost) entirely on Linux. I'm going to tell you how we do that, but first a little history.

It was 2002. I saw a posting on one of the Linux User Group mailing lists from a business asking for some Linux help. That posting led to our first client, but more importantly it proved that there was a market for providing Linux help to businesses.

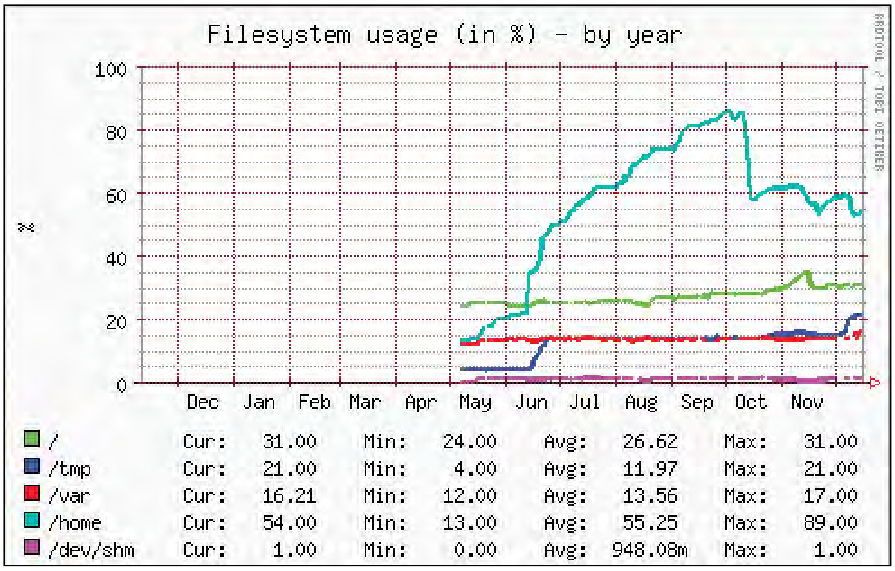
Back then, clients would call from time to time with Linux problems. That was good: I thought that was what we were there for. Those problems would split into two groups, and the first would be requests such as 'can you add a new user?' or 'can you install this software?' or 'could you set up a new server?' These are change requests: part of the IT person's role in ensuring that the IT systems keep up with the business requirements.

The second type of problems would be faults: 'email isn't arriving' or 'we can't save documents' or 'Sally/Joe can't access the Accounts' share'.

There's little we can do about change requests other than fulfil them: we can't know that a new user is starting next Monday unless someone tells us. But we can do something about the faults. We can monitor the mail queues; we can monitor free disk space, and we can monitor a lot more besides.

It's clearly better all round if we can detect that mail is not being sent by the server, investigate the problem and, ideally, resolve it before our client is even aware. This approach was implicitly backed up by our own clients when I asked them what the most important element of the service we supplied was. The answer was always the same: reliability, uptime and availability.

So we started monitoring the servers we supported. There are many open source server monitoring utilities out there; we tried a few and settled on *Nagios*. For those who are unaware of *Nagios*, the basic principle is simple. A central server (the *Nagios* server) contacts each of the servers it's monitoring in turn and instructs them to run a local program or script and feed the results back. The *Nagios* server collates those results and displays them on a web page. Each result falls into one of four categories: OK (displayed with green background), Warning (yellow), Critical (red) or Unknown (indicating an internal problem with *Nagios* itself and displayed in orange).



› ***Munin* is very useful for graphing *Nagios* parameters to track problems and their resolution.**

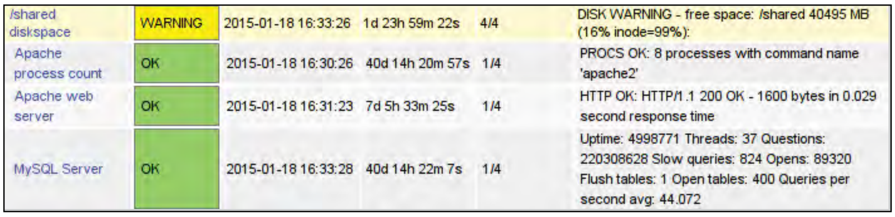
Pre-empting problems

Unlike our clients, we're not interested in what is working, so we filter out all the green lines and just show the rest (*pictured below*). Now we're getting somewhere! The **/var** partition on ServerA becomes 80% full; 80% is the transition we have set between OK and Warning, so the 'how full is your **/var** partition' check now shows up with a yellow background. If that partition gets to 90% full, it will become Critical and go red. At either, or both, stages, an email or SMS notification can be sent to our support staff. They investigate and perhaps resolve a log rotation problem; **/var** drops down below 80% full, the monitor returns to green and life goes on. No service interruption due to a full **/var** partition.

At first, we missed a few things. Occasionally, a client would report a problem that we hadn't been warned about, so then we had two things to fix: the client's server, and our monitoring. We would write an additional check that could be run on that server or, more often, run on many or even all servers. Over time, our monitoring infrastructure grew, and the number of problems we didn't detect in advance fell to close to zero.

Our goal now is that our clients should never have to tell us about server problems: we should know about problems before they do, and we say as much on our website.

So, that's what we do and that's why we do it. Let's look some more at how we do it. »



/shared diskspace	WARNING	2015-01-18 16:33:26	1d 23h 59m 22s	4/4	DISK WARNING - free space: /shared 40495 MB (16% inode=99%):
Apache process count	OK	2015-01-18 16:30:26	40d 14h 20m 57s	1/4	PROCS OK: 8 processes with command name 'apache2'
Apache web server	OK	2015-01-18 16:31:23	7d 5h 33m 25s	1/4	HTTP OK: HTTP/1.1 200 OK - 1600 bytes in 0.029 second response time
MySQL Server	OK	2015-01-18 16:33:28	40d 14h 22m 7s	1/4	Uptime: 4998771 Threads: 37 Questions: 220308628 Slow queries: 824 Opens: 89320 Flush tables: 1 Open tables: 400 Queries per second avg: 44.072

› ***Nagios* enables you to find the 'bad wood' among the healthy trees... or servers.**

The software we use

I mentioned earlier that the business runs almost entirely on Linux, so let's get the aberration out of the way: our accounts system runs on Microsoft Windows.

We'd rather it ran on Linux, but we've not been able to find an accounting system that meets our business needs and runs on Linux (and not for the want of looking). If it's any consolation, the Windows system itself is virtualised and running under Linux.

Our staff are free to choose whichever Linux version and desktop environment they want to run on their desktop PCs; as it happens, we all choose to run Debian with XFCE.

All of our servers run Debian; we use *Exim* for mail; *Apache* for web; *Pacemaker* and *Corosync* for clusters; and *Shorewall* for managing iptables firewalls.

Perhaps unsurprisingly, we use *Libre Office* for all our documents, spreadsheets and presentations. We use *OwnCloud* (hosted on our own servers) for our 'documents anywhere' requirement. We favour Python for scripting, and we use the Python web framework Web2py for a couple of internally-written applications. We use *git* for source code control and *Jabber* with Psi for communicating between ourselves.

Despite trying to use technology for everything, sometimes you can't beat a whiteboard and a handful of coloured pens.

» There are three main ingredients, and the first is system monitoring. Exactly what we monitor on a server is determined by the server's role, but typically there are around 45-50 parameters that are measured, and most of them are checked once every five minutes, 24/7, 365 days a year.

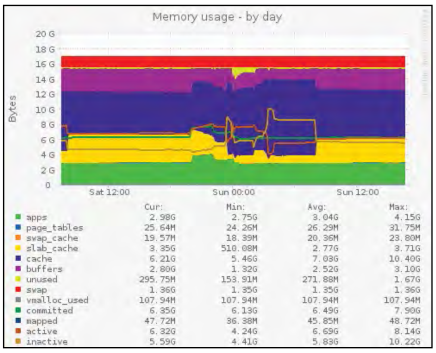
Smart monitoring

As well as the almost real-time monitoring that *Nagios* provides, we also graph a lot of the same parameters using *Munin*. There are a huge number of graphs available. The example graph [*see bottom of page*] shows memory usage. There's a lot of information here, but note that a significant chunk of memory is being used as a disk cache (the dark blue area) to improve disk read times. At the top of the graph, the red area shows swap space used.

Unlike *Nagios*, we don't monitor the *Munin* graphs as a matter of routine; rather, we look at them in the context of solving a problem.

For example, we were alerted (by *Nagios*) to a system where the **/home** partition was 80% full. A quick investigation showed the data taking up the space was just data in the **home** directories, as might be expected.

The Munin graph of disk space used, [*see top, p49*] shows a quite steep but also quite a linear rise. It's easy to see that, if nothing is done, the **/home** partition will be full around the end of October. That information is invaluable: we can call the client, explain what is happening, and also tell them how quickly the problem needs to be resolved. In this particular case, it was easy to find one user who was using the vast majority of the space. Asking them to clean up their **$HOME** directory resolved the problem, as can be seen by the drop in space used in mid-October [*see top, p49*].



	Cur:	Min:	Avg:	Max:
■ apps	2.98G	2.75G	3.04G	4.15G
■ page_tables	25.64M	24.26M	26.29M	31.75M
■ swap_cache	19.57M	18.39M	20.36M	23.80M
■ slab_cache	3.35G	510.08M	2.77G	3.71G
■ cache	6.21G	5.46G	7.03G	10.40G
■ buffers	2.80G	1.32G	2.52G	3.10G
■ unused	295.75M	153.91M	271.88M	1.67G
■ swap	1.36G	1.35G	1.35G	1.36G
■ vmalloc_used	107.94M	107.94M	107.94M	107.94M
■ committed	6.35G	6.13G	6.49G	7.90G
■ mapped	47.72M	36.38M	45.85M	48.72M
■ active	6.32G	4.24G	6.69G	8.14G
■ inactive	5.59G	4.41G	5.83G	10.22G

› **A memory usage graph. Use *Munin* graphs to help solve problems not for monitoring.**

Another element of monitoring involves checking log files. As anyone who has looked through a typical Linux system's logs will know, there is a myriad of information there: a user logs in; an email is received; the internal clock is adjusted by 27 milliseconds etc.

Occasionally, there will be events logged that should be acted upon. Maybe a disk is reporting errors, or perhaps there are repeated attempts to log into a nonexistent user account. The difficulty is in finding the messages that are significant among the thousands of benign messages. The needles are small and the haystack is big: certainly searching the logs manually is both time-consuming and inefficient.

> **Nagios: "The aim here is to only ever receive reports that will be acted upon."**

One approach is to define what is being sought, and have a report sent each time a match is found. The challenge, though, is defining what to look for. Searching for 'error' in the logs might highlight some interesting entries, but it won't find a line reporting 'Unknown user: fredbloggs', for instance

A better approach is to define what we don't want to know about, and then report on everything else. This approach sends emails to the sysadmin detailing everything in the logs that the system hasn't been told to ignore. As you might expect, initially that can be a lot of data, with most of it being benign.

The aim here is to only ever receive reports that will be acted upon: if something is reported that will be ignored, that "something" should be added to the filters so it's no longer reported. The end result will typically be a small number of short reports detailing the log entries that didn't match the 'expected' ones, and which require action. It's that action that increases the security or availability or performance of the server.

After monitoring, the second ingredient is system management. In 2006, we needed to make a change to the firewall on every system we supported. It quickly became apparent that making changes to systems by logging into each one individually was both inefficient and risky: one typo and we could lock ourselves out of a server. We needed to find a scalable way to do it.

Efficient management

We are supporting hundreds of systems that perform a variety of roles for businesses. Some of those servers are running Red Hat, some Debian, some CentOS and so on. The typical system management tasks include:

» Install outstanding security updates on all servers on a 'Business Support' contract except those belonging to client A.

» Remove user B from all servers belonging to client X.

» Ensure *Apache* is installed on all web servers except those marked as 'lighttpd', regardless of distro.

» Do all of the above in a secure, scalable way, complete with an audit trail.

That list is quite a challenge. Even "ensure Apache is installed" hides some complexity. On a Debian system, the command is

```
apt-get install apache2
```

whereas on a Red Hat system, it's:

```
yum install httpd
```

The solution is a configuration management system and we use *Puppet*, which comprises a server (the 'puppet master'), configuration information, and client systems. By default, every half an hour every client ascertains 'facts' about itself and sends them to the puppet master. Those facts will be things like the distro being run, the version; the architecture; the system name etc.

The 'Puppet master' then compiles configuration data from the facts, such as the 'node database', the various *Puppet* 'classes' and it sends that configuration data back to

Get a job

Here are some tips for getting a Linux job:

» **CV** The role of your CV is to get you an interview, not a job. Pay attention to spelling, grammar and relevance. Two sides of A4 is sufficient: cut until you can fit it in that space.

» **Covering letter** Send one, but don't start it 'Dear Sir' or 'To whom it may concern.' Do your homework and find out who it should be addressed to. Again, check spelling and grammar, and then have someone else check it.

» **Love Linux** if you want to get that dream job working with Linux every day, show that you are really, really passionate about Linux (and if you're not, maybe this isn't the right career move for you).

» **Be Smart** I don't mean wear a tie, although that's a great idea for the interview, regardless of the office dress code. Linux isn't always easy, and smart people are always needed. Note: Sending your CV in *Microsoft Word* isn't smart.

» **Get Things Done** On your CV and in the interview, say what you have achieved, not what you were responsible for. Businesses employ people to make them money (hard truth).

» **Linux Support Experience** If you don't have commercial experience, help friends, go to a LUG or offer to support a server for a local charity.

Combine points 'be smart' and 'love Linux' and you'll demonstrate that you could do this commercially, given the chance.

› Keith Edmunds and his team at Tiger Computing have turned a love of Linux into a business.

the client. The *Puppet* agent on the client then enforces the client state to match the configuration data (for example, by installing the package **apache2**).

The node database describes how the client node should look, and there are many ways that this can be implemented. We use a hierarchical structure called, imaginatively, 'hiera'. This data hierarchy is specific to us, and a simplified version of it looks like this:

» **Common data** This applies to all servers, regardless of Linux distro, client etc. One example is that we always disable the Ctrl+Alt+Delete combination (so that Windows users don't reboot the Linux servers by mistake: yes, it has happened).

» **Common data for a specific client** This applies to all servers belonging to that client. Example: setting the text of the 'message of the day' (motd), the welcome text you see when you log into a Linux system.

» **Specific role within client** This might apply to all web servers belonging to client X, and might implement a skeletal holding page with their logo for new websites.

» **Specific site within client** This applies to all of one client's servers at a specific location, and might point those servers at a specific local DNS server.

» **Full qualified domain name (fqdn)** Specific to a server, eg sets the root password.

Detailed below is a sample of a server-specific hiera file (with blank lines removed):

```
---
classes:
  - shorewall
packages:
  - joe
  - samba
root-pw: "$6$4s...IdvYocg7sb1Wf."
```

This system belongs to a class called **shorewall**; packages **joe** and **samba** are to be installed; and a variable called **root-pw** is defined, which holds the hashed value of the root password. Any of those entries could be put into a different file and be applied to all servers belonging to this client. So how are those variables used? That's where the *Puppet* classes come in. Each class comprises a definition of one element of the server setup. For instance, below shows the class applied to all servers, that disables the Control+Alt+Del key sequence:

```
class tclbase::ctrl_alt_del {
    # Disable control-alt-delete
    exec { 'Disable control-alt-delete':
        path    => '/usr/bin:/usr/sbin:/bin',
        command => 'sed --in-place=.bak -e
\'s/^ca:/#ca:/\' /etc/inittab',
        onlyif  => 'grep \'^ca:\' /etc/inittab',
    }
}
```

You can see a simple *sed* command that will edit the ***/etc/inittab*** file, where that key sequence is defined, and comments it out. However, it only comments it out (**onlyif**) if it isn't already commented out. In most cases this class will only take action the first time it runs, but if someone edits ***/etc/inittab*** and uncomments that line, the next time *Puppet* runs it re-enforces the change.

Puppet configuration can be a complex beast. All of the config files are held in a *git* repository, and there are testing frameworks included. We use *Puppet* to manage almost all routine system management tasks, including installing security updates, updating firewall configuration and managing packages.

Nailing process

That's monitoring and management in place; the last ingredient to ensure that our clients never have to tell us about a server problem is processes. One can almost hear a unified sigh: processes? Boring!

Well, that may be true, but look at it this way. We have over 4,000 lines of hiera files and 170,000 lines of class code and associated data in our *Puppet* git repository. Not having some kind of process defining how we organise and update such files is quickly going to cause problem, eg if each of our consultants is free to set up email however he or she likes, the next consultant to look at that server will have to first of all work out how it's been done. So we have standards, processes, documentation. Nothing is cast in stone: if there's a better way of doing something, we want to discuss it – but until we've agreed to change, we do it the current way.

There are exceptions, of course. One client may be using *Postfix* and another *Exim*, but we document how any particular server differs from our norm on the relevant client wiki page.

In summary, then, these are the applications we use to build our client management infrastructure: *Nagios* for system monitoring; *Munin* for trend monitoring; *Logcheck* for log file monitoring; *TWiki* dealing with internal documentation; *Puppet* covering configuration management and *Request Tracker* for the ticketing system.

If you want to read how many of those might work together in a 'typical day', take a look at **http://bit.ly/TigCompRole**. LXF

Why pilots wear uniforms

Does wearing a uniform help a pilot fly the aircraft? Clearly not; if anything, he or she may do a better job in more comfortable clothes. So why do they wear uniforms?

Imagine you're sitting in a plane waiting to go to Barcelona for the weekend. A man gets on the aircraft, aged about 30, not shaved for 3 or 4 days, jeans, scruffy sweatshirt and trainers. Without even consciously being aware of it, you start to form opinions of that person. If he then went into the cockpit and sat down in the driver's seat, you'd probably be quite concerned. Yet nothing in his appearance has the slightest relevance to flying a Boeing 737 safely to Barcelona.

As human beings, we all jump to conclusions on the flimsiest of evidence. Our brain likes to have a full picture of what we're dealing with, even if that picture is wrong. Most IT support staff work remotely, so suppose a user sends a quick email to the support team asking them to look at something, and the reply they get back is poorly put together, perhaps with typos and grammatical mistakes. The recipient will form conclusions, probably unconsciously, about the sender, and may well wonder whether the sender is to be trusted to fix their IT problem.

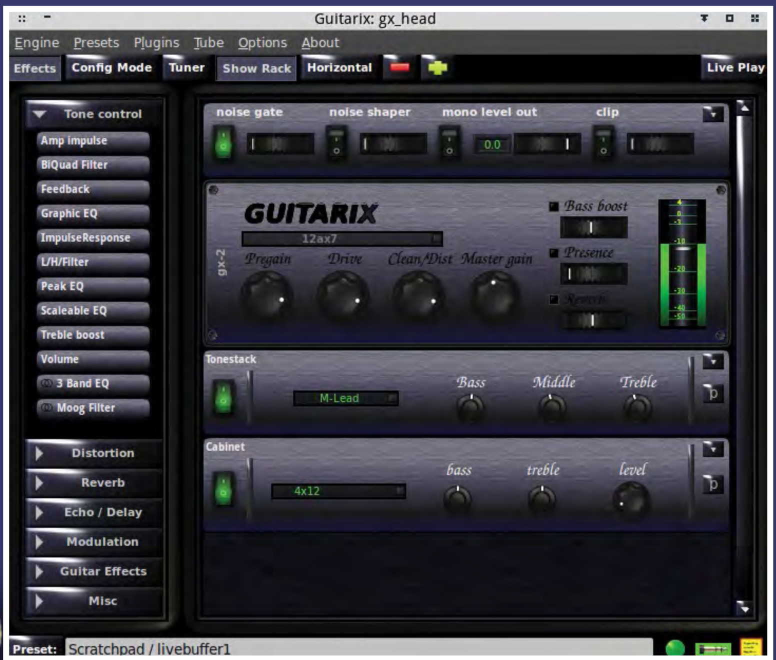
Good IT departments fix the confidence problem by making sure their IT staff can communicate in a clear and literate way.

School of Guitarix

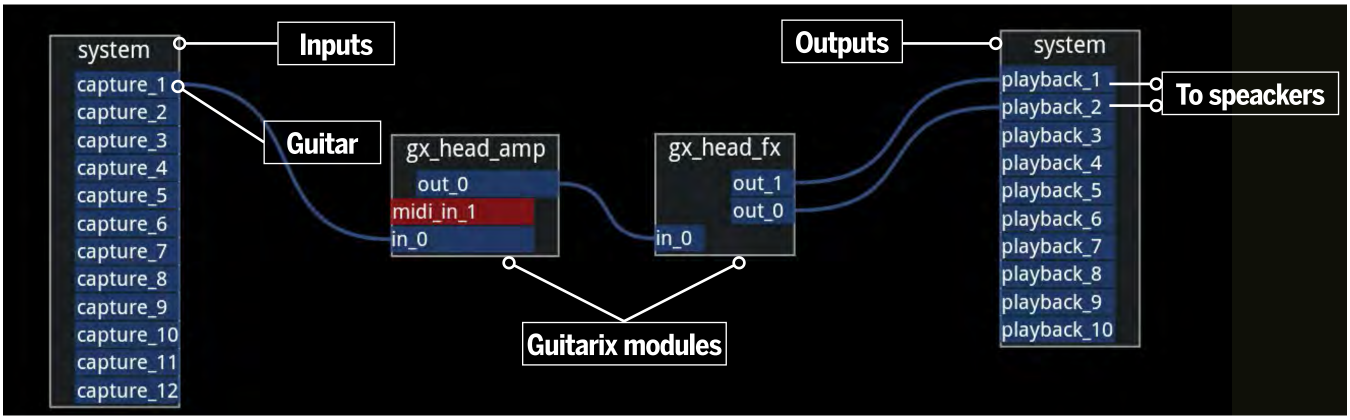
Rock out with **Conor McCormack** as he takes you back to basics and explains pre-amps, MIDI control and effects modules.

G*uitarix* is a modular, virtual amplifier for Linux. With it you can choose different pre-amp and amp models, combine them with various effects and speaker cabinet emulations to come up with your very own tones. *Guitarix* comes as a standalone application, but its modules are also available in the LV2 plugin format, which you can incorporate into your DAW (Digital Audio Workstation) of choice. Furthermore, it can even be run headless, so you can turn a Raspberry Pi, or any other such devices, into a dedicated amp modeler. You can even control *Guitarix* via a MIDI controller or foot-board.

Before you get started, you'll need to know how to correctly get your guitar signal into your computer. First, you'll need to make sure your audio interface is selected by the Jack Audio Connection Kit (JACK). This must be selected and JACK must be started before starting up *Guitarix*. There are various ways to do this but two of the most popular are to use *Qjackctl* or alternatively, *Cadence*, which is bundled with *KXStudio*. To learn more about the fundamentals and getting JACK set up read our recent article [*Features, p48* **LXF191**], but lets plug into *Guitarix*.



For more pro-level music guides, advice and help visit: libremusicproduction.com



› The guitar is connected into input one on the audio interface. It is then routed through *Guitarix* while the sound goes out through output 1 and 2 of the interface and into the speakers.

Your first consideration depends on what type of audio interface you are using. It's important to match the guitars high output impedance level to that of your audio interfaces input or else there will be degradation of the guitar signal, notably, the high frequencies will drop off and you will have a thin, noisy and possibly distorted signal. There are a few ways to do this correctly:

» **Instrument input** If your audio interface has this input, use it. Alternatively, some interfaces have inputs that can be switched between line in and instrument. In this case, make sure you have it set to 'instrument'. Note that on some interfaces, instrument inputs are labeled Hi-Z.

» **DI Box** If your audio interface doesn't have an instrument input, you need one of these boxes. This will correct the impedance mismatch so that you can then connect it up to a line in or mic input.

Next, start up *Guitarix*. When you have your guitar hooked up into your computers sound card, you will need to connect up it up to *Guitarix*. You can use the patchbay area of *Qjackctl* or alternatively more graphical programs such as *Patchage* or *Catia*, to make your connections. For the standalone *Guitarix* application, see our diagram (*above*) for how your input and output connections should look.

Note that *Guitarix* is broken up into two JACK modules. The first one is called gx_head_amp, which is a mono module containing the amp and mono rack effects. The second is a stereo module for stereo rack effects. If these are not correctly connected up, make sure the output of the amp is going into the input of the stereo effects section. You can then treat them as one module in most cases; however, you can insert effects from other JACK applications in between gx_head_amp and gx_head_fx if you wish.

Latency settings

Latency is the length of time that it takes your computer to process a signal and give you back the result. Audio latency is not something that you want if you are playing your guitar live through *Guitarix*. This will result in the sound being played back through your speakers with a slight delay. These delays, while small, can completely throw you off your performance. For this reason, you will want to get your audio latency low enough that the delay is not perceivable. However, this comes at the cost of increased CPU usage.

How low you can get your latency settings depends on both your hardware and software set up. For this purpose, an audio-based Linux distribution is recommended, as these will have a low latency kernel and other optimisations that will enable you to get superior performance over stock Linux distros for low latency audio.

You can adjust your latency (frame rate) in *Qjackctl* or *Cadence* but if you are using the standalone *Guitarix* client, there is an easier way of doing this. If you go to Engine > Latency, you can choose your frame rate setting here. The advantage of changing it from within *Guitarix* is that you can change the setting on the fly, ie there's no need to stop and restart JACK each time you change the setting. It will default to either 1,024 or 512 frames. Those settings will typically cause a noticeable delay. Try lowering the frame rate in increments until you get the audio latency low enough that it is not perceivable. A setting between 64 and 256 will usually give you the best results.

Interface explained

When you first start up *Guitarix*, you will be presented with the amp head and nothing more. You will find some toggle buttons along the top. These do the following:

» **Effects** Toggles the visibility of the effects plugin bar.

» **Config mode** Collapses the height of all effects modules and enables you to easily click and drag around their order. This is useful for reordering large amounts of effects.

» **Tuner** Toggles the visibility of the tuner rack.

» **Show Rack** Toggles the visibility of rack effects modules.

» **Horizontal** This changes the way that the rack modules are displayed and presents them across two columns: the mono »

Convolution

Guitarix includes a convolution module called Convolver that enables you to load up impulse response files, and you will find Convolver under the Reverb category.

With Convolver you can load up reverb impulses although another very good use for convolution is using it as a speaker cabinet emulator by loading up speaker cabinet impulses. Any impulse responses saved in WAV file format can be used with *Guitarix*. A search online will yield lots of forums full of them.

To get you started though, *Guitarix* comes with some guitar cabinet impulse response files preinstalled. To load these up, click on the Setup button in the Convolver module. Next, click on File. You will see two bookmarks on the left called amps and bands. In these folders you will find WAV files that you can load up as impulse responses. If you are using impulse responses for your cabinet emulation, make sure you disable your cabinet module.

» plugins in the left-hand column and the stereo plugins in the right-hand column.

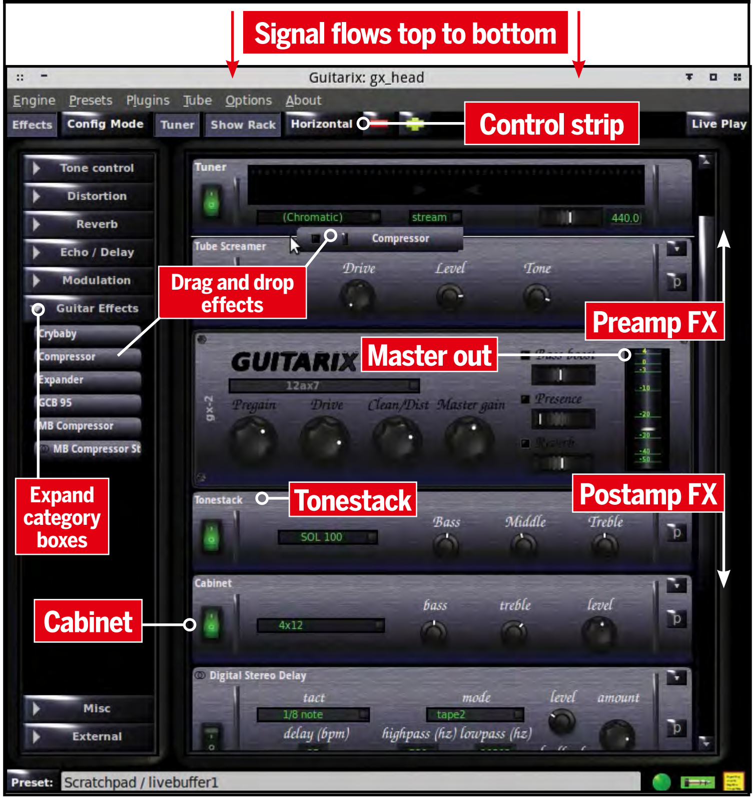
» **Plus and minus buttons** The minus button collapses the height of all rack effects modules, while the plus button expands them again.

» **Live play** This shows a minimal fullscreen display with preset information, which is useful for live performance.

You'll want to start adding and arranging modules and effects. Press the Effects toggle button. You will now see all available plugins and effects down the left-hand side. These are categorised into collapsible boxes. You can expand any of these categories by clicking on a category box.

To add an effect to your signal chain, simply click and drag it across to the rack area and drop it into position. Effects default to disabled when first added to the rack. On the left-hand side of each effect module is a button, which you can toggle to enable or disable the effect. These modules can be moved around in various orders, and can be added before (above) or after (below) the pre-amp section. Just hover over the rack handles on either side of a module, click and drag to rearrange the order. To remove a module from the rack, grab it and drag it back into the plugin bar. The signal is processed through these modules from top to bottom. If you add stereo plugins, they can only be added to the bottom of the rack. This is because stereo effects are processed after mono effects, and hence why *Guitarix* shows up as two separate JACK modules. If you are using the horizontal option for displaying rack effects, the signal flows from top to bottom through the mono rack on the left and then into the stereo rack on the right, again being processed from top to bottom.

Exploring the Guitarix interface



The order of your signal chain is important. The basic structure that will mimic a real amp and speaker cabinet set up needs to have a tonestack and cabinet module below (after) the preamp. You will find both of these modules in the Tone control category:

» **Preamp** Boosts your guitar signal, with tube emulation.

» **Tone Stack** Processes the guitar sound. There are selectable amp models which you can choose

» **Cabinet** Emulates the playback through selectable guitar cabinets.

The amp head controls are as follows:

» **Pregain** The input level of the guitar.

» **Drive** The signal gain.

» **Clean/dist** The amount of distortion.

» **Master gain** The amount of signal sent from the pre-amp to the power section.

» **Master out** This control is an exception to the top to bottom signal flow rule. On this meter you will notice a small white slider. If you drag this up and down it will boost or attenuate the final output signal (to JACK signal). This is useful for attenuating the signal so it doesn't clip on its way out of *Guitarix*.

Effects modules

Guitarix comes with an extensive list of effects, including compression, distortion, modulation, reverb, delay, EQ, etc. Some of the effects modules that are included in *Guitarix* are influenced by some popular hardware units, for example the Tube Screamer is, not surprisingly, based off of the Ibanez Tube Screamer.

As well as its built-in effects, *Guitarix* can also make use of any LAPSPA or LV2 plugins that you have installed on your system. These have to be specifically enabled for them to show up in the plugin sidebar. To do this, go to Plugins > LADSPA/LV2 Plugins. Any enabled plugins will now show up in your plugin side bar under the category 'External'. Between *Guitarix*'s built-in plugins and third-party plugins, there is a vast amount of effects to choose from.

Controlling Guitarix via MIDI

Almost all parameters in *Guitarix* can be controlled via MIDI. This includes changing the values of knobs right through to turning on and off modules and even scrolling through menus.

Before you create any MIDI bindings, you will first need to make sure that your MIDI device is hooked up to *Guitarix*. You can make this connection using any of the many connection tools that Linux has to offer but one method is to use the built-in MIDI connections window. Go to Engine > Jack Ports and click on MIDI tab. You can select your MIDI device here.

Once you have your MIDI device connected up, creating bindings is as simple as pressing the middle mouse button on whichever parameters you wish to control, followed by moving what you want to bind it to on your controller. Being able to control so much this way makes *Guitarix* a very good tool for live use. If you have a MIDI foot-board, for example, the Behringer FCB1010, you can bind your foot switches to turn various effects modules on and off. Alternatively, if your MIDI controller has knobs, you can also bind parameters to these.

The expression pedals can be bound to Wah effects, etc, and your various bindings can be viewed and edited by navigating to Engine > Midi Controller. These bindings can also be saved in presets so that they can easily be recalled for later use.

There are no hard and fast rules for the order to place your effects in. If something achieves the sound you are looking for, then it is the correct way for that purpose, however, there are some generally accepted best practices for the order in which most guitarists prefer to have their effects. The effects table (below) will give you some idea of effects and their recommended order, from top to bottom, in the signal chain.

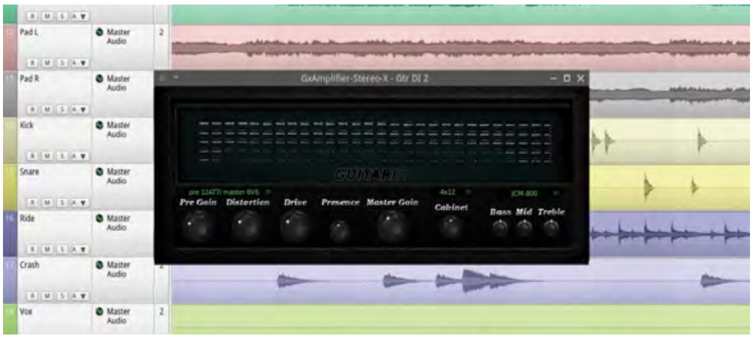
If you press the preset button on the bottom left of *Guitarix*, the preset manager will pop up along the bottom. There are already a few presets here but it's easy to add and manage your own. To create a new preset, click on New in the Scratchpad section. This will prompt you to name your preset. You can then press Save to save your new preset, or tweak it more before doing so. You can also create new banks for your presets and organise them.

Ways to use Guitarix

Guitarix can be used in many different ways: as a practice tool, a recording tool and a reamping tool. Having both the standalone version and LV2 plugin versions, allows for great flexibility of use: as standalone; as a live/live practice tool; to record directly into your DAW or as an insert in your DAW.

Another option is to use the LV2 plugins on a channel strip and record your guitar signal into your DAW dry. This way you have great flexibility in processing your sound later, and tweaking it to perfection, as you will always have the original dry performance.

GxAmplifier is the main guitar LV2 amp modelling plugin (*pictured above*). This plugin consolidates the main amp signal flow mentioned above, into one plugin. So, you have the pre-amp choice on the left, cabinet choice in the middle and tonestack choice on the right. When switching between pre-amp types, the plugins skin changes to a suitable amp type and matching knobs. Additionally, the cabinet and tonestack menus have an off option. This is particularly useful in relation to the cabinet section, as you might prefer to use an impulse response after the plugin for you cabinet emulation. In this case, you'd disable the cabinet section with the off option and add an impulse response plugin after GxAmplifier. We'd suggest two good choices for impulse response plugins, which are IR LV2 and Klangfalter.



› **The main guitar LV2 amp modelling plugin is GxAmplifier.**

As well as the GxAmplifier, *Guitarix* has also ported over many of its effects modules as LV2 plugins. All of these plugins are prefixed by Gx, so if you search for 'gx' in your plugin manager, you will quickly find all the plugins.

Guitarix is a very flexible tool. It can be used in many different types of setups and there are many interesting projects related to it. One project is MOD, a hardware sound processing pedal running *Guitarix* and its various modules. If you want to check out some of the sounds, then head over to the MOD website (**http://portalmod.com/promo**) where you will find some sound demos. LXF

Effect order

EFFECT	RECOMMENDED GUITARIX MODULE	TIPS
	PRE-AMP EFFECTS	
Noise gate	Built-in noise gate	Cuts off low level noise. This is built into *Guitarix* as first in the signal path.
Compressor	Compressor Category – Guitar Effects	Compressors work best when placed before any sound shaping effects.
Overdrive/Distortion	Tube Screamer Category – Distortion	The tube screamer is a popular guitar pedal. It's often used to add a signal boost without driving up the gain setting too much. This hotter signal can drive the pre-amp section harder.
Graphic EQ	Graphic EQ Category – Tone Control	Useful for sculpting your tone, cutting problematic frequencies, mid boost etc.
Wah	GCB 95 Category – Guitar Effects	The GCB 95 is based off of a crybaby wah pedal. If you have a MIDI controlled expression pedal, you can use it to control this.
	POST AMP EFFECTS	
Reverb	Zita Rev1 Category – Reverb	This can add space around your guitar. More suited to clean/lead tones.
Delay (sometimes called Echo)	Digital Stereo Delay Category – Echo/Delay	Delay, or Echo, can often be more suitable than reverb. Whereas reverb can blend a guitar into the back, delay can make a guitars sound more up front.
Convolution	Convolver Category – Reverb	This can be used for adding reverb impulse response files, but also is a good alternative to using the cabinet module if you have impulse response files of cabinets.

Mr Brown's Administeria

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*.

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

Ain't no reboot

When *Linux Format*'s esteemed editor [me? - Ed] first got in contact with me about picking up the Administeria column he handily included a list of previous articles Dr Brown (Sorry, no relation) has penned in the last 80-odd issues he's contributed to. It was rather daunting list to read to say the least! There doesn't seem to be a topic Chris hasn't covered in some way, shape or form in that time. Matching this prolific output combined with the high quality The Good Doctor has managed will be some feat. In some ways, picking up this column feels like walking into a new sysadmin job, except it doesn't come with the one advantage that sysadmins and governments have in common – being able to blame the previous employee when things go wrong!

It does beg the question though – what to cover? It doesn't seem as though the life of a sysadmin is getting any easier; the physical systems of years ago have been replaced by virtual equivalents at a ratio of many multiples to one. Security threats and demands have grown, and the delineation between the traditional sysadmin and other roles, particularly development, is becoming increasingly blurred.

With this in mind, over the next few issues we'll be looking at some open source tools that sysadmins can use in their day to day jobs to make life easier. This month we'll begin with Ansible, the – dare we say fun but definitely – interesting take on configuration management. I'd really like to hear from readers of Administeria with any suggestions of areas they would like to see covered, especially on subjects that cause them problems or that they find confusing.

Please feel free to drop me an email on these or other sysadmin-related subjects. My hope is this excellent column will carry on being as useful and relevant as it has been for me over the years.
jolyon.brown@gmail.com

Development continues apace at CoreOS

The battle with Docker heats up as etcd hits 2.0 and CoreOS targets the modern infrastructure biz.

CoreOS (**https://coreos.com**), the Linux distro, which declared itself production ready halfway through last year, announced that *etcd*, a key component of its infrastructure, had also reached a stable release point. This follows the announcement back in December that it was working on *Rocket*, a container 'replacement' for *Docker*. as the CoreOS team felt *Docker* was diverging from it's original intentions.

The distro itself boasts a very small core, and uses containers for hosting services, and includes a cluster manager (*Fleet*) which it says enables CoreOS to deliver 'easy warehouse-scale computing'. It does this by aggregating individual machines into a single pool of resources. Services which need to be run are submitted to the cluster and *Fleet* decides where they should run. It also allows for graceful updates of the OS across a cluster, handles machine failures automatically and allows for efficient resource utilization.

Etcd is a distributed key/value store which CoreOS uses to share data across it's clusters and it's in use by many other projects (including Google Kubernetes and Cloud Foundry – Pivotal Software's platform as a service). What *etcd* does is provides applications with details on their database connection credentials and enables services to advertise to, say, a proxy server that they are online and are available for routing traffic to.

While being open source, the company behind CoreOS offer a paid 'push' model where software updates are applied without intervention from the local support teams. Despite the danger of this causing aneurysms for some sysadmins and security teams, in this increasingly cloud and centric – some may say hyped – world there's mileage in a service that supplies a modern OS to customers managed in the same way as a browser or retail appliance.

The distro is quietly gaining momentum with the likes of RackSpace and even Redmond starting to offer it on their public clouds (it's an increasingly popular choice on OpenStack clouds). I plan to cover CoreOS in a lot more depth in a forthcoming issue, watch this space.

Ansible for easy config management

Learn how to consistently manage huge groups of machines armed with little more than an SSH connection and some Python.

Configuration management software is now pretty commonplace, but not universal by any means, in any organisation that has to manage multiple machines (virtual or otherwise). There are quite a few choices available in the free/open source world, some of which are now pretty mature. However, one slight criticism often levied against tools such as *Puppet* and *Chef* are that they require their own infrastructure to be useful.

A central 'management' type server is often needed to store the configuration, and agents need to be distributed out across an estate for them to be able to pull settings out to individual nodes. Any additional infrastructure of this kind brings it's own headaches. You'll need to consider backups and/or high availability, as well as bespoke firewall and/or routing changes. So while being able to consistently wield control over a large number of machines with a high server to sysadmin ratio is desirable, the effort needed to get to that point is quite high.

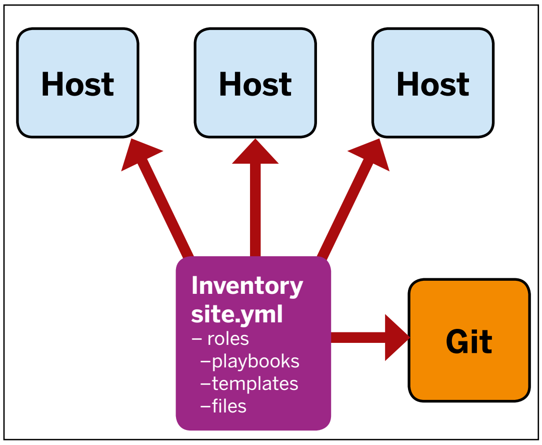
Enter Ansible

Ansible aims to reduce the barriers to using configuration management. It achieves this by having a very minimal set of requirements for getting up and running. Aside from the **ansible** package itself (which will needs to be installed on local machines used by the sysadmin/dev ops teams) most, if not all, the other elements of the *Ansible* stack come bundled with any Linux distribution by default (Python 2.5 or later, and SSH). Rather than using specialised agents to configure target systems, *Ansible* uses SSH connections and nodes need no extra software installation to run.

The actual config steps themselves are written in YAML (**www.yaml.org**), which makes them easy for humans to read, write and more importantly understand. *Ansible* also makes use of a templating language called Jinja2 (**http://jinja.pocoo.org**) based on the templating found in *Django* (another Python-based bit of software) but extends it to make it more powerful.

At the basic level, most work in *Ansible* involves writing a list of tasks to be carried out on a set of target hosts. These tasks, written in YAML, are carried out in the order listed in a file called in *Ansible* lingo as a 'playbook'.

Now lets install *Ansible* and give it a whirl. This is very straightforward, requiring a simple **sudo apt-get install ansible** on Ubuntu or **sudo yum install ansible** on Red Hat-based systems (after enabling the EPEL repository). It's also pretty easy to install from source (or from *Ansible* repos). Installing from *Ansible* will generally mean you get a more up-to-date version of the software (1.8 vs Ubuntu's 1.5).



› **_Ansible_ can use SSH and easy-to-read text files to control all your hosts. Just remember to version control everything.**

Once installed locally on our development machine, we need a target system to configure (we won't cover actually installing any target OSes here). This system could be anywhere: a local VM, a physical box on the network or out in the cloud. It doesn't matter so long as it's reachable by SSH and we have the logon credentials for it. Ideally, using SSH keys is the way to go here. (The command **ssh-copy-id** is worth investigating for distributing keys to boxes you have password access to, and it comes bundled with OpenSSH.)

The first thing to do is to add the hosts details to an inventory. There's a default inventory located under **/etc/ansible/hosts** (but inventories can be created anywhere and multiple inventories can exist (a common use case is to have separate inventories for production and development hosts, for example). We can edit that file and add our target host (and have a look at some of the example lines). Lets assume it's called **testserver.example.com**. Also, we could reference it by it's IP address instead if we wanted.

```
….
# Ex 1: Ungrouped hosts, specify before any group headers.
testserver.example.com
```

One thing to note is that hosts can be grouped together here by simply listing them under a title between two square brackets eg, [webservers]. When you run *Ansible* commands, you can target groups like this, so that installing *Apache* as in the example above on multiple machines becomes just as easy as installing it on one. »

Ansible modules and idempotency

As I've mentioned, *Ansible* comes complete with many modules (a full list of modules is in the docs: **http://bit.ly/AnsibleModules**). These cover a wide range of areas, broadly related to individual tasks a sysadmin might carry out regularly. Some of these are available in an 'extras' module found on GitHub (**http://bit.ly/AnsibleModExtras**). These are expanding all the time and looking down the list will usually throw up things you need – everything from the basic shell command type tasks to managing windows boxes (should you wish to do so). If for some reason what you need isn't available, modules can be extended with locally written ones.

There's a handy guide to doing this at **http://bit.ly/AnsibleDevMods** *Ansible* isn't fussy about the language modules are written in so long as it can handle JSON. Python is popular though, given *Ansible*'s roots

It's important to understand that modules attempt to be idempotent in *Ansible*. This means that they will not try to change anything unless it's required. The idea here is that I could safely run playbooks against my systems over and over in the knowledge that they won't break. This takes a bit of testing and getting used to at first, but is well worth the effort. However a team running *Ansible* (or other types of tools) need to be very wary of manual changes, which aren't pushed back into *Ansible* and version control. The whole team needs to get onboard with a system like this or problems will result and be hard to track down.

» A quick command to show that everything is working is to run a ping against our host. It's worth noting that *Ansible* will attempt to connect to the target using the current username, just the same way SSH would. If you have a different account you want to use on the remote node, specify that with the **-u** parameter to the **ansible** command (which itself is just for running a single command):

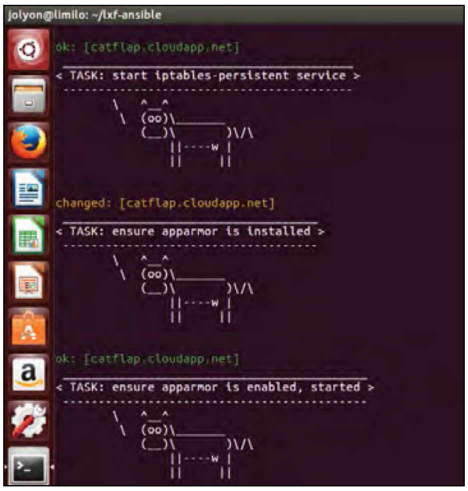
```
~$ ansible all -m ping -u jolyon
testserver.example.com | success >> {
    "changed": false,
    "ping": "pong"
}
```

When it comes to actually implementing change on the target host, there's a fair chance that root level access might be required (but not always). There are a couple of ways to handle this: either use root as the user supplied to the **ansible** command (probably not recommended) or use the **--ask-sudo-pass** flag. You could attempt to run **apt-get update** against our target node here, but that would receive permission errors:

```
 ansible all -a "apt-get update" -u jolyon
```

Using the **sudo** parameter, however, I'll be prompted for a password before seeing the familiar list of apt data-sources and before we return back to the prompt:

```
ansible all -a "apt-get update" -u jolyon --ask-sudo-pass
```



```
jolyon@limilo: ~/lxf-ansible
ok: [catflap.cloudapp.net]
 ________________________________________
< TASK: start iptables-persistent service >
 ----------------------------------------
        \   ^__^
         \  (oo)\_______
            (__)\       )\/\
                ||----w |
                ||     ||

changed: [catflap.cloudapp.net]
 ___________________________________
< TASK: ensure apparmor is installed >
 -----------------------------------
        \   ^__^
         \  (oo)\_______
            (__)\       )\/\
                ||----w |
                ||     ||

ok: [catflap.cloudapp.net]
 _________________________________________
< TASK: ensure apparmor is enabled, started >
 -----------------------------------------
        \   ^__^
         \  (oo)\_______
            (__)\       )\/\
                ||----w |
                ||     ||
```

› **A nice option in *Ansible* is the use of *cowsay* if it's installed and having your orchestration efforts commented on by a herd of bovines.**

Playbooks & roles

Now that I've tested that *Ansible* is communicating with our host, let's turn to actually writing a playbook. Let's assume that our single server is a new host running Ubuntu 14.04 and I want to configure it with some sensible security and system defaults. I want to set a firewall up, and ensure that *apparmor* is installed and enabled. Note: A properly secured server would have not only these steps but many more).

Let's start with creating a new directory to hold our *Ansible* playbook and, more importantly, allow us to hold all of our *Ansible* configuration in source control (I'm assuming you can install *git* locally if you haven't got it already).

```
mkdir lxf-ansible
cd lxf-ansible
git init .
```

Our personal preference is to hold inventory files in the same source-controlled location as the rest of our files. Let's make a sub-directory and create a new inventory file in it (substituting your server name for our example here).

```
mkdir inventory
echo [ubuntu] > inventory/lxf
echo testserver.example.com >> inventory/lxf
```

Here, I've put the server into an 'ubuntu' group. Now, let's create a playbook, using whatever text editor we have to hand. Save it with the filename **lxf.yml**:

```
---
- hosts: ubuntu
  vars:
   - iptables_ports:
     - { protocol: "tcp", port: "22" }
     - { protocol: "tcp", port: "443" }
  tasks:
  - name: Install iptables-persistent
    apt: name=iptables-persistent state=present
  - name: ensure firewall includes our ports
    template: dest=/etc/iptables/rules.v4 src=iptables.j2
owner=root group=root mode=0600
    notify:
     - restart iptables
  - name: start iptables-persistent service
    service: name=iptables-persistent state=started
  handlers:
  - name: restart iptables
    service: name=iptables-persistent enabled=yes
state=restarted
```

OK, time for a quick review of what's happening. The first line specifies that I've targeted the 'ubuntu' group I created just now. I then set some variables for use in the playbook, such as iptables ports I'm looking to enable. The next couple of tasks here should be pretty understandable. The playbook installs the **iptables-persistent** package using *apt*, then drops a templated *iptables* configuration file onto the system. The template line mentions a source file with the name **apache2.conf.j2**. the .j2 means it's a Jinja2 source file (*see below*). The notify line here is a special action for *Ansible*,

Jinja2 templates

The Jinja2 templating system (**http://jinja.pocoo.org/docs/dev/**) is a very powerful element of the *Ansible* stack. It allows files to be generated using data available to *Ansible* at the time the playbook is run. Within the template itself, various control structures and expressions can be used as well as passing variables from *Ansible*. A good place to start here is the Template Designer documentation page on the Jinja website. With a bit of practice, generating configuration files etc for Linux host will becomes second nature.

There are a few tricks worth noting when getting started with templates, though. Knowing which variables are available through the facts-gathering phase of a playbook run, and figuring out what can be gathered from the inventory etc can be a bit frustrating at the beginning. Luckily, these can be dumped out with a special template file. Save a copy of this for future use (call it something like **dumpall.j2**)

```
Module variables ("vars")
--------------------------------
{{ vars | to_nice_json }}

Environment Variables ("environment")
--------------------------------
{{ environment | to_nice_json }}

Group names
--------------------------------
{{ group_names | to_nice_json }}

Group variables
--------------------------------
{{ groups | to_nice_json }}

Host variables
--------------------------------
{{ hostvars | to_nice_json }}
```

This can then be called from a playbook:

```
- name: dump all variables
  template: src=drumpall.j2 dest=/tmp/ansible.all
```

You will then have a nice collection of variable names and output in **/tmp** on your target host.

which says that if the template line changes the file, then the handler to restart *iptables* must be notified. If the file doesn't change, then the handler won't take any action. Finally, we just ensure the *iptables-persistent* service gets enabled to pick up our changes if it hasn't been already.

Now create a second file with the following content and save it as **iptables.j2**:

```
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -m state --state RELATED,ESTABLISHED -j
ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
{% if iptables_ports is defined %}
{% for data in iptables_ports %}
-A INPUT -p {{ data.protocol }} -m {{ data.protocol }} --dport {{
data.port }} -j ACCEPT
{% endfor %}
{% endif %}
COMMIT
```

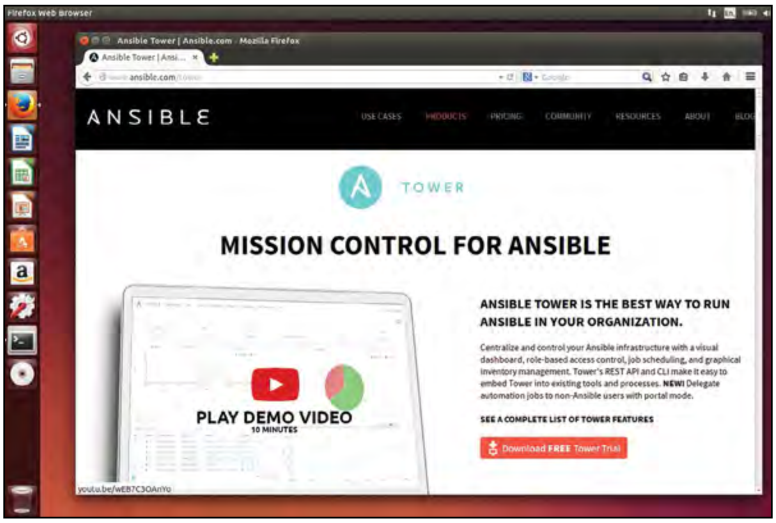
This file is an example of a Jinja2 template. The lines enclosed within **%**s are Python statements, and the values within the **{{ }}** are replaced with values generated at runtime and everything else makes it into the file as is. The template has access to the 'vars' declared in our playbook, namely the **iptables_ports** list of values. These are then looped over and their contents transferred into an *iptables* entry. Variables can be presented to templates in quite a few different ways within *Ansible*, and combined with the templating system provide a very powerful tool for sysadmin use. Once I've made these changes, best practice is to add them to *git* and commit them. I can test this playbook out as follows:

```
ansible-playbook -i inventory/lxf lxf.yml —sudo —ask-sudo-
pass -u <name of user to connect as>
```

This should result in output similar to the following:

```
PLAY [ubuntu] ***********************************************************
*********

GATHERING FACTS ***************************************************
***************

ok: [testserver.example.com]
TASK: [Install iptables-persistent] **********************************
*********

ok: [testserver.example.com]
TASK: [ensure firewall includes our ports] ***********************
*************

changed: [testserver.example.com]
TASK: [enable iptables-persistent service] ***********************
*************

changed: [testserver.example.com]
NOTIFIED: [restart iptables] *****************************************
***********

changed: [testserver.example.com]
PLAY RECAP ************************************************************
************

testserver.example.com       : ok=5    changed=3
unreachable=0    failed=0
```

Note that the first thing *Ansible* does is gather 'facts'. These are in fact (sorry) variables populated by *Ansible* with everything it can find out about the host that it's running against. These are quite detailed. They include a lot of information on the system architecture, IP addresses, processor cores available etc. These can then be used within the playbook (and within templates) as required.



MISSION CONTROL FOR ANSIBLE

ANSIBLE TOWER IS THE BEST WAY TO RUN ANSIBLE IN YOUR ORGANIZATION.

Centralize and control your Ansible infrastructure with a visual dashboard, role-based access control, job scheduling, and graphical inventory management. Tower's REST API and CLI make it easy to embed Tower into existing tools and processes. **NEW!** Delegate automation jobs to non-Ansible users with portal mode.

SEE A COMPLETE LIST OF TOWER FEATURES

› A lot of open source vendors support themselves with 'value-added' products, Ansible Tower is a browser-based application with enterprise friendly features.

Fact gathering can be disabled, which is handy in certain circumstances, eg one off runs against multiple hosts.

After the playbook has run, I run the same command again. This time some of the tasks complete as 'OK' rather than 'Changed'. This is because *Ansible* knows it has no work to do for this run, as the system is already in the desired state.

Logging onto the target system reveals, via the command **sudo iptables -L** that the system now has ports 22 (SSH) and https (443) available, with others blocked off.

Pushing on, I re-open the **lxf.yml** file and add the following lines to the bottom of the 'tasks' section (but above the 'handler' lines).

```
 - name: ensure apparmor is installed
   apt: name=apparmor state=present

 - name: ensure apparmor is enabled, started
   service: name=apparmor enabled=yes state=started
```

And re-run the **ansible-playbook** command again [*see above*]. This should now ensure *apparmor* is available to me on my Ubuntu host. You may notice that our code featured the keywords 'apt' , 'service' and 'template' in the examples. These refer to *Ansibles* built in modules. The software has many of these available in it's core libraries with many more downloadable, and with the option to write my own [*see the Ansible modules and idempotency box, p57*].

While these are relatively trivial examples, I've shown how to create a playbook; chosen which nodes to run it against; dynamically created files containing information within it; and made our server a little more secure. I could easily add to the hosts I want to run this against by adding them to my inventory. Also, I could decide to use the playbook to create a role. Roles are a great way to organise playbooks. By assigning a group of hosts roles, I can ensure they all get the settings they need. So for example, all ubuntu hosts could get my 'security' role that I've just started to develop above. They might also get a 'webservers' role while others get a 'mysql' role. All of these are then referenced in a file usually known as **site.yml**. This is a playbook that describes the whole infrastructure, referencing roles within it. By running **site.ym**l, I could build my whole infrastructure from scratch, and by targeting another inventory, I can replicate my infrastructure on another environment entirely.

This only scratches the surface of *Ansible*, but hopefully I've given you a sense of how straightforward it is to get up and running. There's a lot of information available at **http://docs.ansible.com**. LXF

The best new open source software on the planet

LXFHotPicks

Alexander Tolstoy
The hottest bits of the latest and the greatest open source software around are brought by Alexander for your perusal.

AbiWord » GParted » Handbrake » Stellarium » Gnome Battery Bench » Krita » Albert » Xonotic » X-Moto » TLP » MDP

Text processor

AbiWord

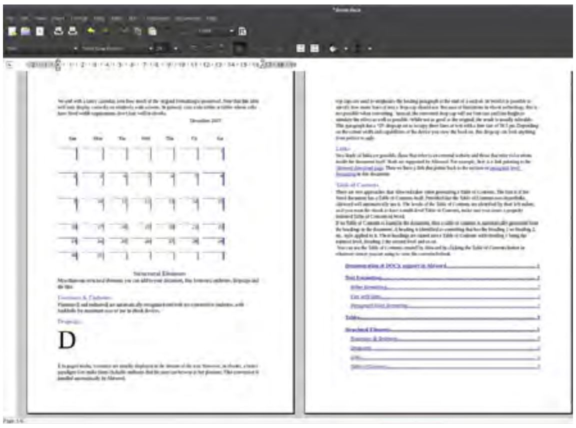
Version: 3.0.1 **Web:** www.abiword.org

We're revisiting *AbiWord* again, after this outstanding word processing software landed in a previous HotPicks [*p60*, **LXF179**]. The new version is 3.0.1, which may seem a minor update compared to 3.0.0, but the *AbiWord* developers haven't been idle for the past year and a half since rolling out that major release. And, again, the team is offering something special for those who are involved in lots of typing and formatting of documents.

The most praised and highly anticipated feature of *AbiWord* 3.0.1 is the default OpenXML plugin, which has better support for the DOCX file format. There has been DOCX support in a few previous *AbiWord* releases, but it was optional and many packages (such as for Gentoo or FreeBSD) disabled the OpenXML plug-in because it wasn't considered stable enough. Now the problems are mostly solved and, when you have to, you can now open *Microsoft Word* documents directly, modify them and save, and write any changes to the original file.

After rolling out the 3.0.0, the *AbiWord* developers faced many stability and performance issues that were introduced along with their much-praised transition to the GTK3 toolkit. In 3.0.1 most of those issues are solved. For instance, *AbiWord* no longer crashes while importing complex documents and redrawing some areas; and it also no longer has GTK memory leaks and is generally more stable.

> “Now has a default OpenXML plugin, and better support for DOCX.”

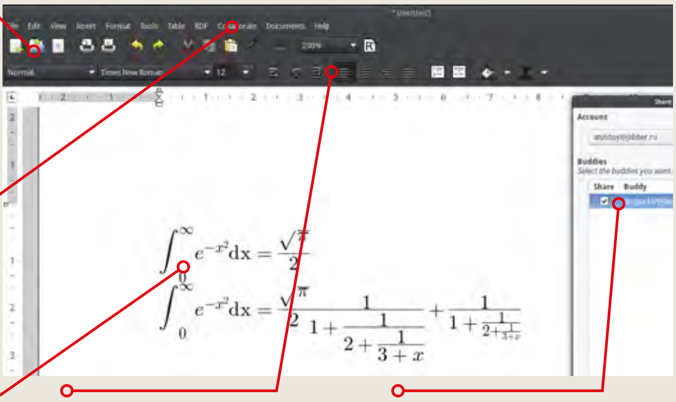


› DOCX support is perhaps not perfect yet but it's still very decent and getting better with each release.

AbiWord remains a feature-rich, yet lightweight competitor of *LibreOffice Writer*, and it offers gorgeous collaboration features, including *Telepathy* support, along with support for the Resource Description Framework (RDF), the original grammar-checking parser (known as Link Grammar) and some handy optional plugins. We were pleased to be able to work with correctly formatted Latex formulas and had fun with *Open Text Summarizer* (*libots*), which automatically extracts the plot from voluminous texts.

AbiWord can be found in almost any Linux distro, but we can't guarantee that the software will be updated everywhere to the latest version, but you can compile the application from source. This way also brings more freedom, because you can decide, which features you want to enable or disable, according to your preferences.

Exploring the Abiword interface


File Edit View Insert Format Tools Table RDF Collaborate Documents Help
Normal
Times New Roman
12
200%
Share
Account
Buddies
Select the buddies you want...
Share Buddy

$$\int_0^\infty e^{-x^2}dx = \frac{\sqrt{\pi}}{2}$$

$$\int_0^\infty e^{-x^2}dx = \frac{\sqrt{\pi}}{2}\frac{1}{1+\frac{1}{2+\frac{1}{3+x}}}+\frac{1}{1+\frac{1}{2+\frac{1}{3+x}}}$$

Familiar toolbar
If you're still missing the classic non-ribbon interface, these convenient buttons are here to stay.

Collaboration
Don't gloss over this powerful feature, as it'll show you how to open access to your document for others to see and alter.

MathML support
AbiWord does a good job when it comes to handling mathematical expressions, equations and formulas.

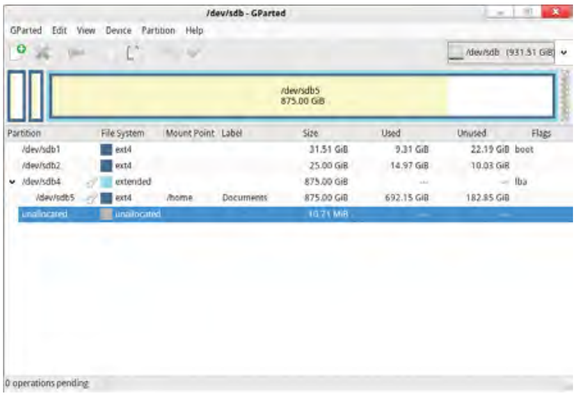
Easy text formatting
Most essential and frequently used features are just one click away. These include numbered and bullet lists, alignment and filling etc.

XMPP-ready
Connect your Jabber account to *AbiWord* and select a buddy that you want to share your document with.

Partition manager

GParted

Version: 0.21 **Web:** http://gparted.org



Partition	File System	Mount Point	Label	Size	Used	Unused	Flags
/dev/sdb1	ext4			31.51 GiB	9.31 GiB	22.19 GiB	boot
/dev/sdb2	ext4			25.00 GiB	14.97 GiB	10.03 GiB	
/dev/sdb4	extended			875.00 GiB	---	---	lba
/dev/sdb5	ext4	/home	Documents	875.00 GiB	692.15 GiB	182.85 GiB	
unallocated	unallocated			10.71 MiB	---	---	

0 operations pending

› **Your disk layouts are easier to alter now that duplicate actions when resizing a partition have been removed.**

GParted is a free partition editor and a neat graphical application for managing disks and partitions. The name actually stands for GNU Parted, as it was initially started as a GTK front-end for the console **parted** command.

GParted isn't the official Gnome partition manager, the official one is *Disks* aka *Palimpsest Disk Utility*, but we really think that *GParted* is more beloved and valued among Linux community. The new release, which was rolled out after half a year of development, brings many promising and attractive features.

The key changes in *GParted 0.21* include a fix for an off-by-one sector error with editor's internal block copy; support for Ext4 file-systems on RHEL/CentOS 5.x; and the removal unnecessary duplicate actions when resizing a partition. *GParted* also takes care to support enterprise-level Linux distros that are already aging, even when RHEL 5.x and its libre derivatives use legacy 2.6.18 kernel and thus are unable to mount Ext4 partitions, you'll still be able to create them with *GParted*, which is lovely.

Another game changer is the inclusion of ReFS support. ReFS stands for Resilient File System and is from Microsoft. It's intended to become the default filesystem in a future post-NTFS Windows. ReFS was introduced in Windows Server 2012 and uses B+ trees for on-disk data structures, allocation-on-write updating of meta-data, and supports some of NTFS features. Last but not least, the new release brings the Reiser4 filesystem back to life, enabling the creation of Reiser4 partitions on modern Linux 3.x kernels. Reiser4 continues to progress on Linux with filesystem patches recently updated for Linux 3.16 kernel support, as well as support for SSD discard option.

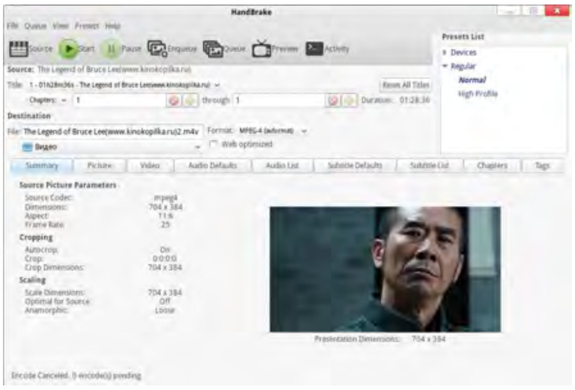
> **"Enables the creation of Reiser4 partitions on Linux 3.x kernels."**

GParted is present in almost any distro, plus you can download a *GParted* Live image (which is based on Debian Sid, **http://bit.ly/GPartedLive**), which you can burn to CD or write to a USB stick. The technology is really awesome, keeping in mind that *GParted* itself has tiny footprint of just 2MB.

Video transcoder

Handbrake

Version: 0.10 **Web:** https://handbrake.fr


HandBrake
File Queue View Presets Help
Source Start Pause Enqueue Queue Preview Activity
Reset All Titles
Presets List
Devices
Regular
Normal
High Profile
Summary Picture Video Audio Defaults Audio List Subtitle Defaults Subtitle List Chapters Tags

Source: The Legend of Bruce Lee(www.kinokopilka.ru)
Title: 1 - 01h28m36s - The Legend of Bruce Lee(www.kinokopilka.ru)
Chapters: 1 through 1 Duration: 01:28:36
Destination
File: The Legend of Bruce Lee(www.kinokopilka.ru)2.m4v Format: MPEG-4 (avformat)
Видео
□ Web optimized

Source Picture Parameters
Source Codec: mpeg4
Dimensions: 704 x 384
Aspect: 11:6
Frame Rate: 25
Cropping
Autocrop: On
Crop: 0:0:0:0
Crop Dimensions: 704 x 384
Scaling
Scale Dimensions: 704 x 384
Optimal for Source: Off
Anamorphic: Loose
Presentation Dimensions: 704 x 384

Encode Canceled. 0 encode(s) pending

› ***Handbrake* supplies more metadata than other tools and while your movie converts, you can explore the file details.**

For years we've been using *FFmpeg*, *mencoder* and *vlc* for various tasks, from extracting audio tracks out of movies to downsizing video clips for use on mobile devices. *Handbrake* is specifically a video transcoder, but it's an enticing option to many convenient encoders and here's why: The program originally served as a DVD ripper, but as time has gone on, it's gained more features, including support for H.264/265, Blu-Ray and DVD subtitles. However, the most prominent feature is a handy bunch of predefined target presets for popular devices. *Handbrake* allows you to seamlessly convert your video – whether local file or optical disc – including Android, iPad and AppleTV. The pane on the right-hand side of the *Handbrake* window houses a tree of those formats. When you click one, the settings are applied instantly in the main window area, but it's possible to choose custom options at any time.

The output settings area have six tabs, of which the most important are Filters and Video. The first one enables you to quickly fix common picture issues, such as telecining (distorted frames of moving objects, when recorded from PAL or NTSC source), interlacing, noise and blocks. *Handbrake* filters are quite effective, though they slow the encoding a little. The Video tab has video codec options for H.264/265, MPEG-2/4, VP8, Theora; container selection (MP4 or MKV) and various codec tunables, such as optimisation rate, framerate selection and even more options for the H.264/264 codec.

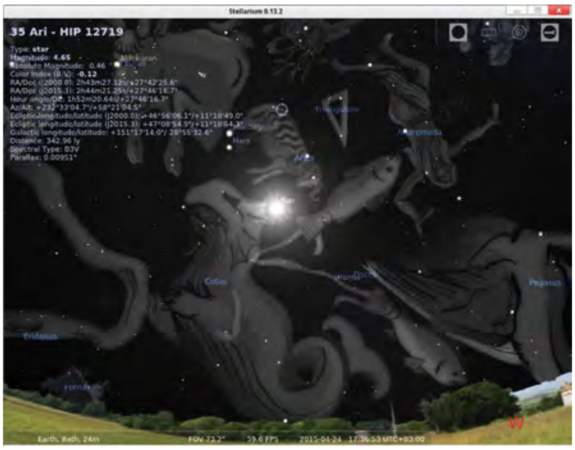
Using *Handbrake* is a pleasure to use – the program is simple, yet powerful and stable. *Handbrake* relies on dozens of open source libraries and codecs, including *FFmpeg* and *GStreamer* plugins. Some components are statically linked and in the *Handbrake* install – the team say it uses custom patches that aren't part of the upstream versions for better stability. More than that, almost its major features are available via *HandbrakeCLI* executable – very useful for scripting.

> **"A pleasure to use – the program is simple, yet powerful and stable."**

Planetarium

Stellarium

Version: 0.13.2 **Web:** www.stellarium.org



35 Ari - HIP 12719
Type: **star**
Magnitude: **4.65**
Absolute Magnitude: 0.46
Color Index (B-V): **-0.12**
RA/Dec (J2000.0): 2h43m27.12s/+27°42'25.6"
RA/Dec (J2015.3): 2h44m21.25s/+27°46'16.7"
Hour angle/DE: 1h52m20.64s/+27°46'16.7"
Az/Alt: +232°33'04.7"/+58°21'04.5"
Ecliptic longitude/latitude (J2000.0): +46°56'06.1"/+11°18'49.0"
Ecliptic longitude/latitude (J2015.3): +47°08'54.9"/+11°18'54.2"
Galactic longitude/latitude: +151°17'14.0"/-28°55'32.6"
Distance: 342.96 ly
Spectral Type: B3V
Parallax: 0.00951"

› Take a trip into the virtual night sky and view 210 million stars from any spot on Earth and beyond.

If you love a flower that lives on a star, it is sweet to look at the sky at night", wrote Antoine de Saint-Exupéry in *Little Prince*. And for Linux users that are no less romantic – but not so eager to pry themselves away from their seats – *Stellarium* is the cure. It's a detailed OpenGL-powered 3D sky emulator, which you can scroll, zoom in and out of and travel to any spot on Earth and beyond.

Stellarium has a catalogue of more than 210 million stars, of which nearly 600,000 are bundled within a local installation. You can gaze at them all with the 'naked eye', which makes sense only for great planets, the milky way and some constellations, or by using binoculars or an even more powerful telescope.

Little and distant stars are rendered as simplified circles, but solar system planets and the Moon are shown with their appropriate textures, correctly lit and with the correct rotation just as if you were looking at them through in real sky. *Stellarium* enables you to switch the atmosphere on and off, and when it is on you can enjoy very detailed sunrises and sunsets on Earth or look at the Earth from another planet or position in space.

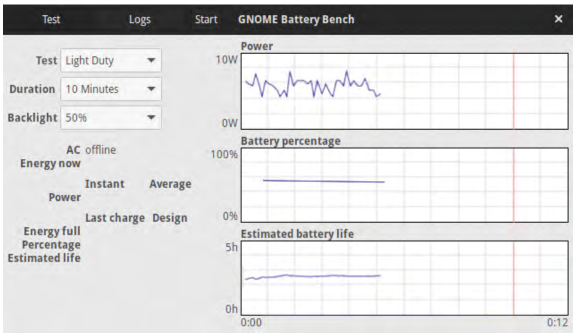
Stellarium is very feature-rich with lots of interface controls, 20 sky cultures and translations in 133 languages. Starting from the 0.13 series, *Stellarium* has started using Qt5 controls, and brings significant improvements over 0.12 version. These include new sky cultures (Arabic, Japanese and Siberian); a more realistic Milky Way, nebulas, comets, asteroids and exoplanets; Zodiacal light; native planets naming for non-English locales and an updated plug-ins stack. Yes, hardcore astronomy nerds will be happy with the new coordinates measuring tool, observatory plugin, telescope control tool and son on. If you find it thrilling, don't miss the new *Stellarium* release and go download the latest version from the project website. The program requires cmake and some **Qt5-devel** packages, which should be available in your distribution.

> “Zoom in and out of a detailed OpenGL-powered 3D sky emulator.”

Battery monitor

Gnome Battery Bench

Version: 3.15.4 **Web:** http://bit.ly/GnomeBatteryBench



Test: Light Duty
Duration: 10 Minutes
Backlight: 50%
AC offline
Energy now
Instant Average
Power
Last charge Design
Energy full
Percentage
Estimated life

› Gnome Battery Bench is a nice and visually appealing front-end to the Powertop tool.

Those who run Linux on a laptop are probably aware of some basic tips, which help improve battery life. Among a bounty of optimising and analysing tools there's the *Powertop* utility for measuring power consumption and generating reports. *Powertop* doesn't take over control of your settings, but analyses them and helps find culprits for battery drain. *Gnome Battery Bench* is a graphical front-end for *Powertop* that controls battery usage in real time, which can be very useful if you want to measure the results of your changes.

The program offers two test cases - Idle and Light Duty. For the latter it plays back recorded sequences of events in a loop and monitors battery charge to estimate power usage. There are three graphs for: power, battery percentage and estimated battery life. Test sequences are editable, plus you can create your own one by running **gbb record** on the command line. The output is saved in a plain text file in **/usr/share/gnome-battery-bench/test**, which can be played back using **gbb play /your/file.txt**. You can place your own recording in similar files in the folder, which you'll be able to access via a drop-down menu in *Powertop*. For any test you choose you can specify the duration of the loop (5-10-30 minutes or until 5% of battery charge) and backlight percentage (5-50-100%).

Gnome Battery Bench only emerged earlier this year and requires GTK 3.14 or later. Currently it's only packaged for Ubuntu 15.04, but you can build it from source, which is simple if you have X11, GTK3, Evdev and a few other Gnome-related dependencies installed. It's worth noting that the app can be successfully compiled with older GTK3 versions, but just won't run. Also, make sure that you have *Powertop* installed, as *Battery Bench* doesn't check for it.

> “A front-end for Powertop that controls battery usage in real time.”

Raster graphics editor

Krita

Version: 2.8.7 **Web:** https://krita.org

Let's start with an answer to a common question: *Krita* means 'chalk' in Swedish as the program was designed primarily for drawing from the outset. This isn't evident to everyone, as we keep being asked whether *Krita* is an *Adobe Photoshop* alternative or a KDE-friendly equivalent to *Gimp*. The simple answer is: not really. *Krita* is a sophisticated and professional drawing software, targeted at creating hand-drawn digital art. It, therefore, has dozens of templates for strip cartoons, but very few image filters.

Krita is a part of the popular but still lesser-known office suite *Calligra*, and is perhaps the most prominent and complete part of *Calligra*, as the suite, in general, doesn't even come close to *LibreOffice* in terms of features.

But *Krita* is often regarded as a unique, standalone software, and the editor is respected and widely used by digital artists in many communities, such as Deviant Art.

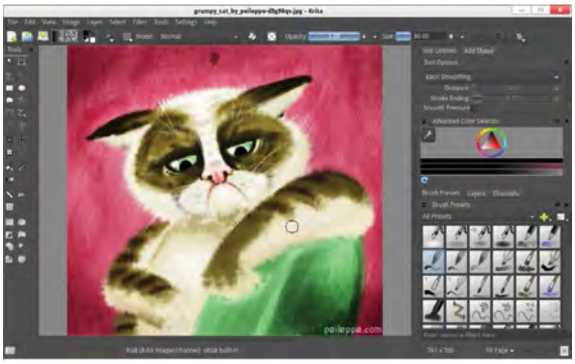
The layout of the main window would be familiar for most users. The toolbar is on the left while colours, brushes, tool options, layers and other stuff are on the right. The top menu houses common file-handling options, such as New, Save, Undo arrows and also Gradients and Filling tools.

For those who explore *Krita* more deeply will find some stunning features. Along with more common tools, such as path drawing and multi-brush painting, *Krita* offers CMYK support, HDR painting, perspective grids, dockers, filters, painting assistants and much more. The sequence of actions, for instance, can be recorded to a macro and played back later.

But the best way to get the most out of *Krita* and see what's possible is to surf through the official gallery of fantastic artwork created in the editor (**https://krita.org/features/gallery**).

Krita is available for most Linux flavours, thanks to its parent *Calligra Suite* package. If you're not sure that you're going to get the latest *Krita* version, you may want to try to compile it from source, but keep in mind that it will be quite time-consuming, as the editor's code is massive.

> “The editor is respected and widely used by digital artists.”



› Even Grumpy Cat can't believe how many brush presets that *Krita* has to offer the digital artist.

Desktop launcher

Albert

Version: 0.6 **Web:** http://bit.ly/AlbertLauncher

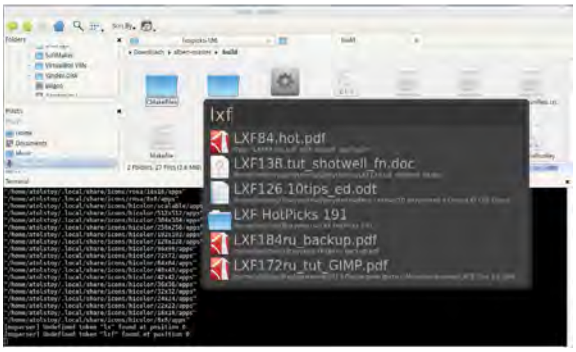
Apple introduced Spotlight over 11 years ago, an amazing system-wide desktop search feature, which has inspired many similar open source technologies. For instance, you might have heard of *Beagle*, which stopped development in 2009, or some more recent and active projects, such as Unity's search lens and *Synapse*. *Albert* is another one application, which is something in between a launcher and a global search tool. It's a fast, Qt5-powered and desktop environment agnostic omni launcher (it's name is a play on the *Alfred* launcher in OS X).

Albert can run applications, open files, open *Chromium* bookmarks, calculate math expressions, search the Internet and more if you enable optional modules. *Firefox* bookmarks aren't indexed right now, but its support is planned for the nearest future. Ubuntu users can get it from **ppa:nilarimogard/webupd8**, other users will likely have to build it from source (you'll need the **Qt5-dev** stack and cmake).

The first time you run *Albert*, you'll be greeted with its Settings window. The first thing we advise doing is to assign a hotkey for invoking the *Albert* search bar. Under the General tab you can also edit the history depth; the number of proposals; and the search bar look, position and extra action modifiers. If you add more folders to *Albert*'s index, please keep tracking the memory usage as while the default *Albert* configuration needs 9-10 MB of RAM, adding a folder with a massive set of files can push memory usage to hundreds of megabytes.

The Modules tab currently has five modules, for web search, calculator, app index, bookmark Index and file index. Each (aside from the calculator) is configurable: For example, you can add or remove search paths or define a path to browser bookmarks file. Another big feature is fuzzy search, which has to be enabled manually for the app index module. *Albert* is under heavy development, and its search backend is constantly evolving, so it's likely that by the time you read this its capabilities will be even more advanced.

> “A fast, Qt5-powered and desktop agnostic omni launcher.”



lxf
LXF84.hot.pdf
LXF138.tut_shotwell_fn.doc
LXF126.10tips_ed.odt
LXF HotPicks 191
LXF184ru_backup.pdf
LXF172ru_tut_GIMP.pdf

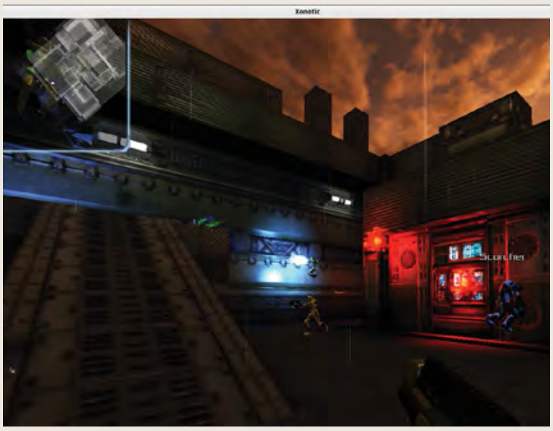
› The concept of a 'search as you type' feature has got faster and better looking thanks to Albert.

HotGames Entertainment apps

First-person shooter

Xonotic

Version: 0.8 **Web:** www.xonotic.org



› **You could stand and stare at the explosions in Xonotic, but you're going to be quickly fragged.**

Xonotic is a fork from *Nexuiz*, an open-source first person shooter that hit controversy in 2010. *Nexuiz* was licensed to Illfonic Game Studios in an attempt to go commercial and Team Xonotic was born. The game runs on a heavily modified version of the DarkPlaces engine from *Quake* and borrows elements from *Unreal Tournament*. Playing this futuristic arena shooter really gets the adrenaline pumping, even if you just select a single player mode and compete with AI bots. The game has many playing modes, but the most relevant are deathmatch and capture the flag.

The previous version of *Xonotic* was released almost two years ago, so the 0.8 release was highly anticipated. Its features include new power-ups (or buffs), three extra maps for capture the flag mode and a brand-new weapon called the Arc, which is a medium-range hitscan gun limited by its risk of overheating. Also, for the first time there are monsters of five types and a special invasion game mode.

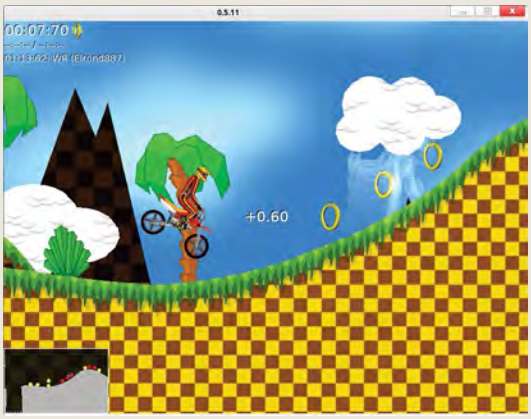
And, of course, there are plenty of performance improvements. After installing *Xonotic 0.8* (a 900MB data package), you'll find that the game creates two launchers: one for OpenGL and another for SDL mode. SDL support isn't a new thing, but the game builds are now optimised with SDL 2.0 in mind. You'll also find OpenGL has less problems with third-party input devices, while SDL provides much greater mouse sensitivity and some visual improvements. We can happily report that *Xonotic* runs very smoothly even on integrated video, while still looking good. You can grab a copy from your package manager or download a build from the website.

> “A futuristic arena shooter that really gets the adrenaline pumping.”

Motocross platform game

X-Moto

Version: 0.5.11 **Web:** http://xmoto.tuxfamily.org



› **Accelerating too fast can flip your rider over, which is fatal in Xmoto and in real life too.**

Thanks to the world-wide adoption of the Android mobile platform, almost anyone has heard of *Hill Climb Racing*, a popular 2D platform scroller. Some of you may have heard of *Elasto Mania* too, which is where the idea for *X-Moto* came from, accompanied by some thoughts on physics emulation.

X-Moto is meant to reflect 'realistic' and super-sensitive control of a motorcycle. The game uses the SDL framework and features very basic cartoonish graphics, accompanied with retro electronic music. As the game is designed for desktop computers (Linux, FreeBSD, OS X and Windows are supported), the controls are very simple: The Up arrow accelerates the cycle, the Down arrow brakes while the Left and Right arrows define the movement. You can also flip the bike instantly by pressing the Spacebar.

The game has a wealth of levels, but very few are included by default, but the first time you run *X-Moto*, the game will insist on connecting to the internet, where there are nearly 2,800 levels available. The levels are divided into packs, which are sorted by things like complexity and theme etc. In each pack you can start with a random level, or try to complete them in order. The goal of the game is to collect items (strawberries, coins etc) and reach an end marker (usually a flower or ball).

The game is heavily oriented to online; each level will show you a global best time ghost. When you have passed the level at least once, there will be another ghost showing your best time. We have to admit, that most levels are hard to complete, so it's recommended to start with the Classical pack. This offers 42 levels with more basic maps and steep terrain, and like a lot of these precision, time trial games, it does a good job of keeping you hooked.

> “Like a lot of time trial games, it does a good job of keeping you hooked.”

Power management tool

TLP

Version: 0.7 **Web:** http://linrunner.de/en/tlp/tlp.html

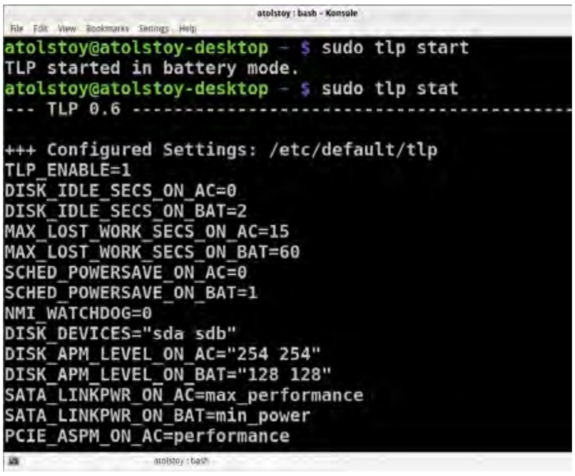
While *Gnome Battery Bench* enables you to silently witness your battery drain, we demand more control. In the past there were dozens of tips, tricks, hacks, tunables and sometimes controversial suggestions that could really squeeze an extra hour from a Linux laptop. But why not make things easier? The first two 'one stop' optimisers were *Jupiter* and *Laptop-mode-tool*, and are both considered as legacy solutions now. The successor is *TLP*, which is an advanced and modern power management tool for Linux that doesn't require you to understand every technical detail.

TLP comes with a default configuration already optimised for battery life, so you may just install and forget it. Nevertheless, *TLP* is highly customisable to fulfill your specific requirements. The tool is also widely available across most major Linux distros (although absent in Mageia). You'll need two packages: **tlp** and **tlp-rdw** (Radio Device Wizard). *TLP* will automatically start upon the next reboot, but if you can't stand the wait (or your laptop fans are roaring dangerously), issue the

```
sudo tlp start
```

command. *TLP* has two modes: for AC and battery respectively. The modes are switched automatically when you change a laptop's power source. *TLP* manages your CPU power state and auto-sets its frequency (something like 'on-demand' in *CPUFreq*), the number of active cores and threads. It also spins down your SATA hard drive, auto-suspends USB (everything but input devices) and applies supported power management policies to things such as the optical drive bay, audio chip, KMS, PCI bus, Wi-Fi and Bluetooth chips and a whole lot more.

It's worth mentioning that *TLP* is incompatible with *laptop-mode-tools*, so you must choose only one solution to manage your power settings. At the same time, *TLP* works comfortably with *Powertop*, because the latter is an analysing tool and not a manager.



```
atolstoy@atolstoy-desktop ~ $ sudo tlp start
TLP started in battery mode.
atolstoy@atolstoy-desktop ~ $ sudo tlp stat
--- TLP 0.6 ---------------------------------------------

+++ Configured Settings: /etc/default/tlp
TLP_ENABLE=1
DISK_IDLE_SECS_ON_AC=0
DISK_IDLE_SECS_ON_BAT=2
MAX_LOST_WORK_SECS_ON_AC=15
MAX_LOST_WORK_SECS_ON_BAT=60
SCHED_POWERSAVE_ON_AC=0
SCHED_POWERSAVE_ON_BAT=1
NMI_WATCHDOG=0
DISK_DEVICES="sda sdb"
DISK_APM_LEVEL_ON_AC="254 254"
DISK_APM_LEVEL_ON_BAT="128 128"
SATA_LINKPWR_ON_AC=max_performance
SATA_LINKPWR_ON_BAT=min_power
PCIE_ASPM_ON_AC=performance
```

atolstoy : bash - Konsole
File Edit View Bookmarks Settings Help
atolstoy : bash

› Now we're confident that our laptop won't misspend precious watts and run smooth and silent.

> **"An advanced, modern power management ºtool for Linux."**

Presentation tool

MDP

Version: GIT **Web:** https://github.com/visit1985/mdp

When you prepare to talk to an audience and need a presentation, filling it with shiny graphics and fancy effects can be a tempting idea. Think twice: simply decorated text with limited colours may work better, especially for technical information. If you agree – and are fed up of slides of cute kittens – *MDP* is an undiscovered gem you've been missing.

In Hotpicks there's always some room for another command-line software alternative, and this time we'll try to make a real presentation using Markdown and *MDP*. Lets start with obtaining the tool. As far as we know, no *MDP* packages exist, but the manual compile procedure is a very straightforward thing. You'll only need essential build tools and **libncursesw5-dev** (or similarly named) package. Then just issue:

```
git clone https://github.com/visit1985
/mdp.git
make && make install
```

And you're nearly done. Before you can launch *MDP*, you'll want to enable some more fancy colour effects, by issuing the:

```
export TERM=xterm-256color
```

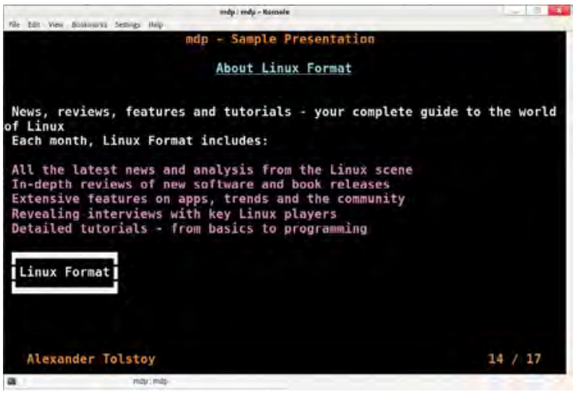
command, which you can also put in your **.bashrc** file.

The tool accepts MD files as input parameters, so the:

```
mdp sample.md
```

command will open the sample presentation, which comes bundled with the tool. The inside of an MD file is plain Markdown text, which supports many formatting options, such as headlines, code, quotes, text decoration and highlighting (bold, underline etc) and UTF-8 special characters. You can even draw pseudo-graphic boxes, using characters such as ┌, ┬, ┐ and so on (see this Unicode chart **www.unicode.org/charts/PDF/U2500.pdf**).

When you finally master your presentation and launch it, use PgUp/PgDn or respective arrow keys to go to the next or previous slide, Home/End keys jump to the first or last slide, and q exits. More keys and examples are available at the project's Git page. LXF



```
                    mdp - Sample Presentation

                       About Linux Format


 News, reviews, features and tutorials - your complete guide to the world
of Linux
 Each month, Linux Format includes:

 All the latest news and analysis from the Linux scene
 In-depth reviews of new software and book releases
 Extensive features on apps, trends and the community
 Revealing interviews with key Linux players
 Detailed tutorials - from basics to programming

 ┌─────────────┐
 │Linux Format│
 └─────────────┘



  Alexander Tolstoy                                             14 / 17
```

› Help your presentation audience concentrate by using *MDP*'s simply decorated text, instead of lots of eye-candy.

> **"Make yourself a real presentation using Markdown and MDP."**

LINUX FORMAT Back issues » Missed one?

Issue 195
March 2015

Product code:
LXFDB0195

In the magazine
All the details on the new 500% faster Raspberry Pi 2! Get started with Linux: a guide to the new distros and how to install them. Port your code to Python 3. Scripting languages explored and setting up a Ghost blog.

LXFDVD highlights
Ubuntu 14.10, Fedora 21, ArchBang 2015.01 and more!

Issue 194
February 2015

Product code:
LXFDB0194

In the magazine
What's the Next Big Thing in Linux OS tech? We show the hot stuff you'll want to try. Bored of your default desktop? Take your pick of our alternatives. Plus, cake for everyone! Firefox celebrates 10 years.

LXFDVD highlights
Fedora 21 Workstation, Manjaro, ALT Linux, 4MLinux and more!

Issue 193
January 2015

Product code:
LXFDB0193

In the magazine
Create a multimedia hub for your home and stream films, music and photos around the house. Try out next-gen filesystems for a RAID array, mod *Minetest* and simplify your firewalls and so much more!

LXFDVD highlights
Ubuntu 14.10, OpenSUSE 13.2 and XBMCbuntu 13.2 and more.

Issue 192
Christmas 2014

Product code:
LXFDB0192

In the magazine
More power! Charge up your distro with essential tips and tools. Build a robot and a monstrous 24TB NAS box. Plus: system recovery, Linux certs and our pick of the most productive desktops.

LXFDVD highlights
Ubuntu 14.10 Remix (pick from 5 desktops), ROSA, Rescatux.

Issue 191
December 2014

Product code:
LXFDB0191

In the magazine
Take your Raspberry Pi mastery to the next level with our hot hacks. Learn how to contain everything with *Docker* and plug in to professional audio production using *JACK*. Plus: Develop with PHP.

LXFDVD highlights
Hot Pi distros! Kali Linux, Jasper, RetroPie, Pi MusicBox and more.

Issue 190
November 2014

Product code:
LXFDB0190

In the magazine
Origin of the distro – **LXF** tracks the source of your favourite distro and picks the best of each genus. Plus: we chart Ubuntu's bumpy history as it celebrates 10 years. Also, Pi alternatives and the best web browsers.

LXFDVD highlights
Tails 1.1 Live DVD, Deepin 2014 and 3 essential rescue distros.

To order, visit www.myfavouritemagazines.co.uk

Select Computer from the all Magazines list and then select Linux Format.

Or call the back issues hotline on 0844 848 2852 or +44 1604 251045 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

Available on the App Store

amazon apps Available at amazon

GET THE MAGAZINE ON Google play

nook

zinio™

*Free Trial not available on Zinio.

LINUX FORMAT **Tutorials**

ntpd Understanding system time, creating config files and querying ntpd

ntpd: How time ticks on Linux

Like the white rabbit **Sean Conway** is late! Here's how to configure the Network Time Protocol and keep the time daemon accurate.

Our expert

Sean Conway is a former electronic technologist in aviation now turned senior system specialist in telecoms.

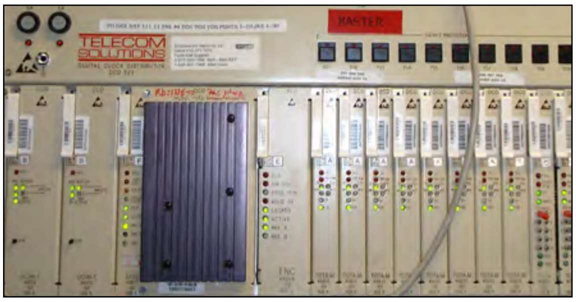
The keeping of time and insuring it's accurate is a common function on computer systems. It occurs under the hood of your system and is a given when establishing servers. The responsibility of ensuring the foundational infrastructure is in place to support time on computers is the systems administrator. This month we'll explore the configuration options to allow the network time protocol (NTP) to manage time. We'll also introduce a tool to examine statics from time sources and cover a few troubleshooting exercises using network tools.

Using the block diagram [*see bottom, p70*], we start our discussion with the hardware box, moving to sources of time and finish with more in-depth details on management – but let's begin with the BIOS and hardware clocks.

If you've ever set up the BIOS you will have been presented menu selections for establishing date and time. This establishes the clock time on the PC's motherboard. This clock answers to many names, a complimentary metal oxide semiconductor (or CMOS) clock, a BIOS clock, a motherboard clock or as in our drawing a hardware clock.

The CMOS battery keeps the clock ticking when there's no AC power to the board. In times of old the clock and battery may have been contained in a CMOS module that plugged into a 16-pin chip socket.

The hardware clock is, in turn, the source of time for the system clock that's maintained by the operating systems. When an OS starts, the system clock seed time is taken from the hardware clock and the OS then takes over. Both of these clocks operate independently and are not synchronised, which means it's possible to reboot and discover the system time isn't accurate because the hardware clock is a poor timekeeper. How far the hardware clock time drifts can result in problems, which we'll discuss later.



› **The GPS time-source receiver and Cesium clock are in the module (to the left of middle) with the black heat sink.**

Managing time

The system clock can call for assistance to get the time right using a network time protocol daemon (ntpd). This daemon sends a request to more accurate external time source for time checks, and using the data provided the daemon feeds the system clock a drift factor to keep the time in sync with the external sources.

One of the limitations of ntpd is how far out of whack the system clock and the external sources are allowed to be. If the hardware clock and system clock differ by approximately <>1K seconds the daemon assumes one of the two time keepers is misconfigured and the daemon will fail.

Since the hardware clock is the system clocks seed time, it becomes important for the hardware clock to be within the <>1K limits. One mechanism to compensate for the hardware clock is *ntpdate*. The program *ntpdate* uses an external time source to provide a one-time adjustment to the systems clock for accuracy.

From the command line you can provide *ntpdate* with a time source. Note: Make sure no time service is running:

```
ntpdate 0.pool.ntp.org
```

On some OSes, before the ntpd starts *ntpdate* uses the external time source to provide a one-time kick to the systems clock for accuracy. The list of time source URLs or IPs for *ntpdate*, look in the **/etc/ntp/step-tickers** file. When ntpd takes over and provides drift compensation it isn't faced with a system clock that's too far out it can't pull it back on time. The hardware clock can continue to run a muck or get some help, but we'll discuss that a little later.

The world of external time sources is divided into levels or stratum followed by a number of accurate time. The closer to the originating source of time, the lower the stratum number. Getting time from a source that's directly connected is considered stratum 0. If that time has been provided to other distribution points, which in turn passes the time on to other points, the stratum number is incremented to a maximum of 15 (See **http://bit.ly/NTPRulesOfEngagement**).

Checking ntp exists

Check to see if a time keeper program is installed and running. Here are some suggestions on how to accomplish the task.

Using the command line, query the OS software manager, eg *yum* or *apt-get*, to determine if the time software is installed. You can scan running processes to *grep* for the network time protocol (NTP) pattern. You could also use the *systemctl* or service tool to determine program status. Additionally, you can cull directories with a **whereis** looking for the program or pursue directories in search of files you know will be there if the program exists.

Providing an example of the best way to achieve the result, requires a clear understanding of the subjective word best.

If the support for NTP is not installed on your system use the software managing tool installation command. Two flavours of the OS software manager commands are provided below for Red Hat:

```
sudo yum install ntpd
```

and for a Debian variant:

```
sudo apt-get install ntp
```

Check the time daemon isn't running with either

```
sudo systemctl stop ntpd
```

or

```
sudo service ntp stop
```

For the start of this tutorial you'll want to rename the original time configuration file to something else. The file can then be created with a configuration for working through this tutorial:

```
mv /etc/ntp.conf /etc/ntp.conf.org
```

The source of most accurate times are atomic clocks, which can be carried in satellites or be ground-based. Satellites have multiple atomic clocks on board, to ensure a more accurate time, and the global positioning system (GPS) requires the time accuracy to put a plane on a runway and not off in the bush [*see bottom, p68*]. Ground-based atomic clocks have been a source of time in the telecommunication industry for decades. Accurate time is important to ensure information transmitted as bits can be decoded as acceptable bits when received at the other end.

NTP config files

That wraps up the three blocks in our hardware, managing and sources drawing (*see bottom, p70*). Lets circle back around for a sequel to managing time by opening a terminal for some more management. Using what ever text editor you were weaned on edit/create a file called ***/etc/ntp.conf***. We're going to create a file that has all the configuration options needed but commented out. We'll remove the octothorpe (pound sign #) and the text Line <number> as instructed, to expand the options in the configuration file. It's important to save the file after each change. Using a second terminal window, we'll run commands to start the time daemon and probe the services workings with query tools. By adding the configuration options line by line, the reader will witness how the daemons operation is impacted:

```
server 0.pool.ntp.org
#Line 1   restrict default kod nomodify notrap nopeer
noquery
#Line 2   restrict -6 default kod nomodify notrap nopeer
noquery
#Line 3   restrict 127.0.0.1
#Line 4   restrict -6 ::1
#Line 5   server 127.127.1.0
#Line 6   fudge 127.127.1.0 stratum 10
#Line 7   driftfile /var/lib/ntp/drift
```

Now make sure to save the file. We'll come back to the config file a few times to remove comments and save. We'd suggest keeping the file open and do a save from one terminal window. If you are running SElinux in enforced mode, the new file will also need the proper context. This is done by referencing the original config file:

```
chcon –reference=/etc/ntp/ntp.conf.org /etc/ntp.conf
```

Start the time daemon using either

```
sudo systemctl start ntpd
```

or

```
sudo service ntp start
```

Let's do some probing using the command-line time query tool *ntpq*. This program is a utility used to monitor the NTP daemon's (ntpd) operation and performance. The commands to enter can be found in the screenshot from the output of our test system [*see top, p70*]. The **-n** switch enables a DNS lookup of **0.pool.ntp.org**. The URL used in our example is a front for a pool of servers. Don't expect to get the same IP shown in our output as it will vary. Just after start it will look like this:

```
[root@fedora19]# ntpq -n -c peers
    remote          refid       st t when poll reach   delay   offset
jitter
==============================================================
==========================
 142.137.247.109 209.51.161.238   2 u    2   64    1   43.446
3.550   0.000

[root@fedora19]# ntpq -c assoc
ind assid status  conf reach auth condition  last_event cnt
==============================================================
========
  1 58621  9024   yes   yes  none    reject   reachable  2
```

After 4 minutes, it will look like this:

```
[root@fedora19]# ntpq -n -c peers
    remote          refid       st t when poll reach   delay   offset
jitter
==============================================================
==========================
*142.137.247.109 209.51.161.238   2 u    4   64   17   43.300
4.534   0.838

[root@fedora19]# ntpq -c assoc
ind assid status  conf reach auth condition  last_event cnt
==============================================================
========
  1 58621  963a   yes   yes  none  sys.peer    sys_peer  3
```

The commands were issued once when the daemon was started and again four minutes later. The line around the condition field indicates the source is being used as a peer. The line around the remote field is to draw attention to the asterisk. This indicates that the daemon has successfully peered with the source for time input. For more information on the *ntpq* tool, the de facto resource for all documentation is on the University of Delaware website found here: **http://bit.ly/NTPQueryProgram**.

What do you do if the daemon is not peering with a source? This is a good time to break out the packet sniffer to »

» see if our daemon is functioning. This will enable us to check if the daemon is sending out a time request and receiving a reply. You can confirm the port number used by *ntp* or other services examining the **/etc/services** file. [*See the red No.1 in the screenshot for the command output, right*]:

```
sudo tcpdump -i <ethernet interface> port 123
```

The output of **tcpdump** displays the daemons call to the external time source every minute. There doesn't appear to be any replies being received. A quick check of the local firewall reveals a closed port 123 on the firewall. Look at the same **tcpdump** output, a reply was received once the port was opened. If you don't see requests being sent then the issue rests with the daemon; it may not have started. Check your install and configuration file.

Securing ntpd

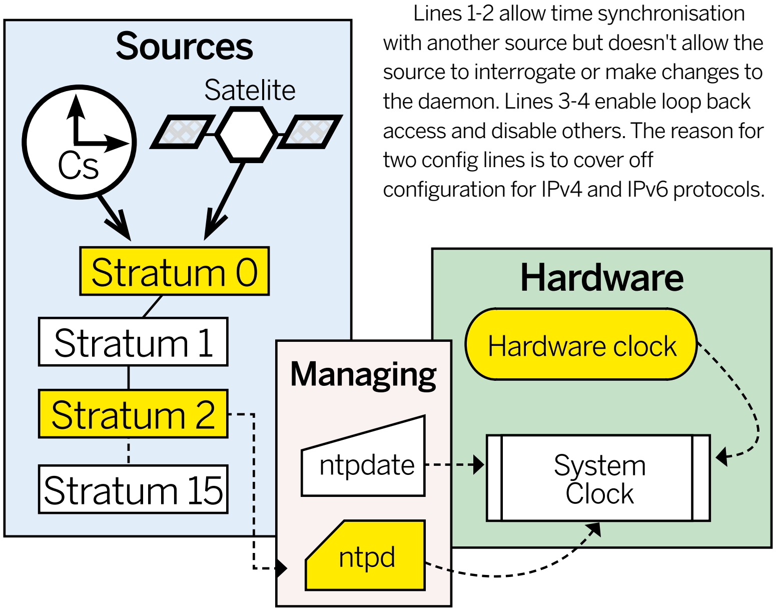
The ntp daemon by default is open and very chatty when queried (*see the green No.2 in screen shot for output, right*). This information enables others computers to get the status and more details. Reading the output is a simple way to tell what kernel a system is running. The security conscious may want to curtail the daemon propensity to speak out. Edit the **/etc/ntp.conf** configuration file by removing the # and Line 1-4 text. You should now have a total of five lines. Don't forget to save the file after making the changes. Restart the daemon to read the configuration changes using

```
sudo systemctl restart ntpd
```

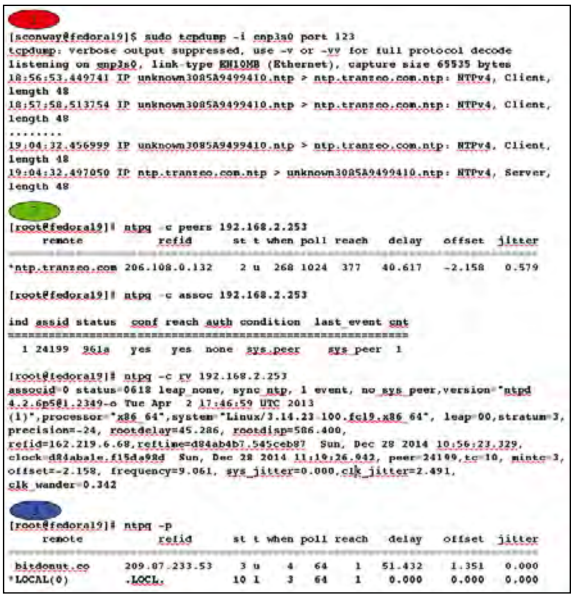
or

```
sudo service ntp restart
```

Lines 1-2 allow time synchronisation with another source but doesn't allow the source to interrogate or make changes to the daemon. Lines 3-4 enable loop back access and disable others. The reason for two config lines is to cover off configuration for IPv4 and IPv6 protocols.



› How time is set, managed and synchronised.



```
[sconway@fedora19]$ sudo tcpdump -i enp3s0 port 123
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp3s0, link-type EN10MB (Ethernet), capture size 65535 bytes
18:56:53.449741 IP unknown3085A9499410.ntp > ntp.tranzeo.com.ntp: NTPv4, Client,
length 48
18:57:58.513754 IP unknown3085A9499410.ntp > ntp.tranzeo.com.ntp: NTPv4, Client,
length 48
.........
19:04:32.456999 IP unknown3085A9499410.ntp > ntp.tranzeo.com.ntp: NTPv4, Client,
length 48
19:04:32.497050 IP ntp.tranzeo.com.ntp > unknown3085A9499410.ntp: NTPv4, Server,
length 48

[root@fedora19]# ntpq -c peers 192.168.2.253
     remote           refid      st t when poll reach   delay   offset  jitter
==============================================================================
*ntp.tranzeo.com 206.108.0.132    2 u  268 1024  377   40.617   -2.158   0.579

[root@fedora19]# ntpq -c assoc 192.168.2.253

ind assid status  conf reach auth condition  last_event cnt
===========================================================
  1 24199  961a   yes   yes  none  sys.peer    sys_peer  1

[root@fedora19]# ntpq -c rv 192.168.2.253
associd=0 status=0618 leap_none, sync_ntp, 1 event, no_sys_peer,version="ntpd
4.2.6p5@1.2349-o Tue Apr  2 17:46:59 UTC 2013
(1)",processor="x86_64",system="Linux/3.14.23-100.fc19.x86_64", leap=00,stratum=3,
precision=-24, rootdelay=45.286, rootdisp=586.400,
refid=162.219.6.68,reftime=d84ab4b7.545ceb87  Sun, Dec 28 2014 10:56:23.329,
clock=d84aba1e.f15da98d  Sun, Dec 28 2014 11:19:26.942, peer=24199,tc=10, mintc=3,
offset=-2.158, frequency=9.061, sys_jitter=0.000,clk_jitter=2.491,
clk_wander=0.342

[root@fedora19]# ntpq -p
     remote           refid      st t when poll reach   delay   offset  jitter
==============================================================================
 bitdonut.co     209.87.233.53    3 u    4   64    1   51.432    1.351   0.000
*LOCAL(0)        .LOCL.          10 l    3   64    1    0.000    0.000   0.000
```

› Packet sniffing to see what ntp's chatting about.

Running the same query commands as before, the daemon isn't responding to requests. Edit the **/etc/ntp.conf** configuration file by removing the # and Line 5-7 text, and restart the daemon to read the configuration changes.

Lines 5-6 enable the daemon to use the hardware clock as a time source, if external sources aren't available. Line 7 enables the daemon to record the hardware clocks drift from system clock in a file. This information assists the daemon on power down restarts. A simple ntp print query displays the local time source in the output [*see blue No.3 in the screenshot above*, for the output]:

```
mv /etc/ntp.conf.org /etc/ntp.conf
```

Using a text editor examine **/etc/ntp.conf** for commented entries. If you plan to use this config file for your installation, you'll need to add your server selections and/or remove the default entries.

Let's finish off back at the hardware clock. The hardware clock values are read with

```
hwclock -r
```

and written from the system clock to the hardware clock with:

```
hwclock -w
```

The man pages suggests running the write command periodically to compensate for the hardware clock drift.

There you have it, taking care of business and working over time; an examination of Father Time provided by the ntp daemon. The *ntp* program follows the client server model and can be used to set time and be used to distribute time as a broadcast. Those tidbits of time distribution configuration you can discover in the original configuration file. LXF

Sensei Conway & The Art of Sysadmin

When being taught (grasshopper), it's important for any junior sysadmin to realise that there's generally more than one way to achieve a given result. This will often depend on a few things: the age of the teacher; the teacher's level of experience on one or more OSes; and their level of patience. And there will be surprising differences.

When a teacher prefaces an instruction with the words "a quick way to do this is…", the listener needs to translate this as how that individual does it, rather than this is the way to do it. There will often be quicker or more efficient ways to the same results. The pupil's job is to take what's offered and build on it and find other ways themselves. When you become the teacher then you share what you've learned for others to build on. A friend suggested to me that everyone has 300 things they can do well. Certainly, there might be some overlap and some unique skills, but we should aim to share our 300 with others and, in turn, learn 300 ourselves so we can grow.

TorBox Use a Raspberry Pi and Tor to set up an anonymising hotspot

Tor: Set up a Wi-Fi hotspot

Mayank Sharma configures a Raspberry Pi as an access point that routes all traffic over the anonymous Tor network.

Our expert

Mayank Sharma has configured so many of his devices for anonymous use recently that even he's confused about who he is anymore. It's all gone a bit A Scanner Darkly.

Do you use *Tor* to prevent big brother from tracking you online? Although it is pretty straightforward to use, it can be quite a hassle to configure *Tor* on all your Internet-enabled devices. You can save yourself a lot of hassle by using a Raspberry Pi as an anonymised wireless access point. The Pi will dole out an IP address and any device that's connected to it will be able to access the Internet via the *Tor* network. To get this project up and running, you'll need a Raspberry Pi along with an SD card with the Raspbian distro. If you haven't done this before, follow the walkthrough to get Raspbian up and running. You'll also need an Ethernet cable. Hook one end into the Pi's Ethernet port and the other into your wireless router. This is how the Pi will connect to the Internet. You'll also need a USB Wi-Fi adaptor that's compatible with the Raspberry Pi. If you haven't got one yet, check the list of compatible adapters that are known to work on the Pi (**http://elinux.org/RPi_USB_Wi-Fi_Adapters**).

Access Point Pi

Once you've setup the Pi, you can configure the Pi from a remote machine via SSH. For the rest of the tutorial, we'll assume the IP address of your Pi is **192.168.2.100**. Fire up a terminal that's connected to the same router as the Pi and enter

```
ssh pi@192.168.2.100
```

to connect to it. After authenticating yourself into the Pi, use

```
iwconfig
```

to make sure the wireless adaptor is recognised by the device. Now refresh its package list with

```
sudo apt-get update
```

and install the software that will make it act as an access point with:

```
sudo apt-get install hostapd isc-dhcp-server
```

When it's installed, it's time to set it up. Begin by editing the **/etc/dhcp/dhcpd.conf** file that controls the DHCP and automatically assigns IP addresses to all connected devices. Open it in the *nano* text editor with

```
sudo nano /etc/dhcp/dhcpd.conf
```

and comment out the following two lines by adding a # in front of them, so that they read:

```
#option domain-name "example.org";
#option domain-name-servers ns1.example.org, ns2.example.
org;
```

In the same file, scroll down and uncomment the word authoritative; by removing the **#** in front.

Then scroll down to the end of the file and add the following lines:

```
subnet 192.168.12.0 netmask 255.255.255.0 {
range 192.168.12.5 192.168.12.50;
option broadcast-address 192.168.12.255;
option routers 192.168.12.1;
default-lease-time 600;
max-lease-time 7200;
option domain-name "local";
option domain-name-servers 8.8.8.8, 8.8.4.4;
}
```

In these lines we define the IP address of our Pi access point (192.168.12.1), the range of the IP addresses it'll hand out to connected devices (from 192.168.12.5 to 192.168.12.50) as well as the address of the domain name servers (8.8.8.8 and 8.8.4.4). You can change any of these values as per your preference. Save the file (Ctrl+X) once you're done.

Setting up a static IP

We'll now edit the **/etc/default/isc-dhcp-server** to specify the interfaces that our new DHCP server should listen to. Open the file and scroll down to the line that reads **INTERFACES=""**. Insert **wlan0** between the quotes so that it now reads **INTERFACES="wlan0"**, and save the file.

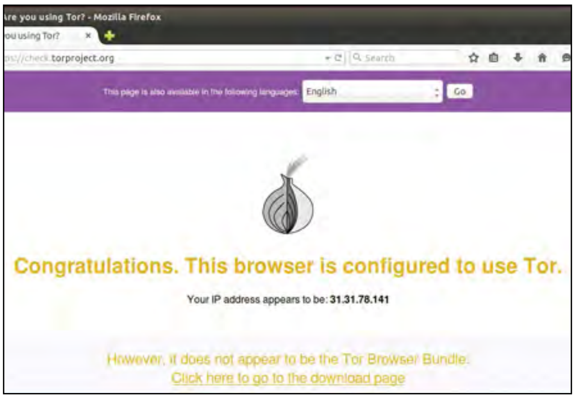
Now we'll setup the wireless adaptor (wlan0) and give it a static IP address. First, deactivate the wireless adaptor with:

```
sudo ifdown wlan0
```

command and then open the **/etc/network/interfaces** file. In the file, comment out every existing entry associated with wlan0, such as:

```
# iface wlan0 inet manual
```

»



Are you using Tor? - Mozilla Firefox

This page is also available in the following languages:

Congratulations. This browser is configured to use Tor.

Your IP address appears to be: **31.31.78.141**

However, it does not appear to be the Tor Browser Bundle.

Click here to go to the download page

› **It takes more than *Tor* to stay anonymous. Make sure you read the documentation on the Tor Project's website.**

Quick tip

If you get Locale errors when connected to the Pi remotely, make sure you don't forward your locale by editing **/etc/ssh/ssh_config** and commenting out the SendEnv LANG LC_* line.

```
# wpa-roam /etc/wpa_supplicant/wpa_supplicant.conf
# iface default inet dhcp
```

Then add the following lines below the line that reads **allow-hotplug wlan0** to set the static IP address for the new access point:

```
iface wlan0 inet static
address 192.168.12.1
netmask 255.255.255.0
```

Save the file and activate the interface with

```
sudo ifconfig wlan0 192.168.12.1
```

Make your point

Now that we've defined the wireless access point it's time to configure it. Create a new file called ***/etc/hostapd/hostapd.*****conf** with the following contents:

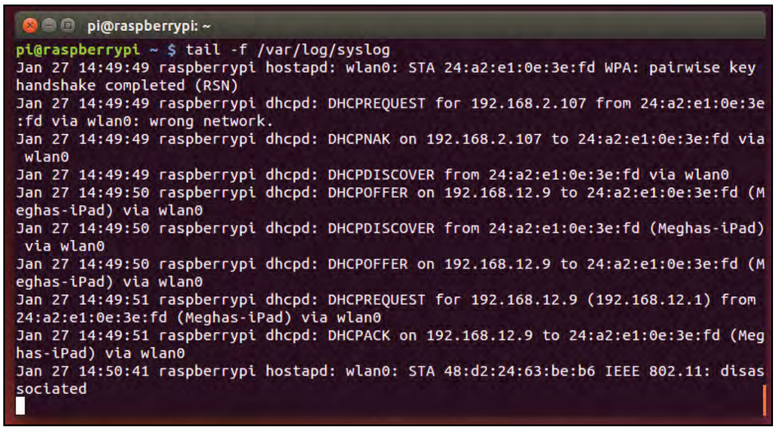
```
interface=wlan0
ssid=TorSpot
hw_mode=g
channel=6
macaddr_acl=0
auth_algs=1
ignore_broadcast_ssid=0
wpa=2
wpa_passphrase=$$Your_Passphrase$$
wpa_key_mgmt=WPA-PSK
wpa_pairwise=TKIP
rsn_pairwise=CCMP
```

We've setup a password-protected network called TorSpot. You can specify a different name for the access point by specifying it in the **ssid=** string. Also change the **wpa_passphrase=** string to specify a custom password. You'll need to enter this password to authenticate yourself to the Pi's access point.

Next up, we'll tell the Pi where to find this configuration file by pointing to it in the ***/etc/default/hostapd*** file. Open the file, find the commented out line that reads **#DAEMON_CONF=""** and uncomment and edit it to read **DAEMON_CONF="*/etc/hostapd/hostapd.conf*"**.

Quick tip

Use the **tail -f / var/log/syslog** command to keep an eye on all system messages. This might come in handy if you are unable to connect to the Pi hotspot.



```
pi@raspberrypi ~ $ tail -f /var/log/syslog
Jan 27 14:49:49 raspberrypi hostapd: wlan0: STA 24:a2:e1:0e:3e:fd WPA: pairwise key
handshake completed (RSN)
Jan 27 14:49:49 raspberrypi dhcpd: DHCPREQUEST for 192.168.2.107 from 24:a2:e1:0e:3e
:fd via wlan0: wrong network.
Jan 27 14:49:49 raspberrypi dhcpd: DHCPNAK on 192.168.2.107 to 24:a2:e1:0e:3e:fd via
 wlan0
Jan 27 14:49:49 raspberrypi dhcpd: DHCPDISCOVER from 24:a2:e1:0e:3e:fd via wlan0
Jan 27 14:49:50 raspberrypi dhcpd: DHCPOFFER on 192.168.12.9 to 24:a2:e1:0e:3e:fd (M
eghas-iPad) via wlan0
Jan 27 14:49:50 raspberrypi dhcpd: DHCPDISCOVER from 24:a2:e1:0e:3e:fd (Meghas-iPad)
 via wlan0
Jan 27 14:49:50 raspberrypi dhcpd: DHCPOFFER on 192.168.12.9 to 24:a2:e1:0e:3e:fd (M
eghas-iPad) via wlan0
Jan 27 14:49:51 raspberrypi dhcpd: DHCPREQUEST for 192.168.12.9 (192.168.12.1) from
24:a2:e1:0e:3e:fd (Meghas-iPad) via wlan0
Jan 27 14:49:51 raspberrypi dhcpd: DHCPACK on 192.168.12.9 to 24:a2:e1:0e:3e:fd (Meg
has-iPad) via wlan0
Jan 27 14:50:41 raspberrypi hostapd: wlan0: STA 48:d2:24:63:be:b6 IEEE 802.11: disas
sociated
```

Use the tail -f */var/log/syslog* common to keep an eye on the devices connected to your *Tor* hotspot.

NAT setup

We now need to set up NAT to allow multiple clients to connect to the Pi's access point and route all their traffic through the single Ethernet IP. Edit the ***/etc/sysctl.conf*** file and at the bottom add the following line:

```
net.ipv4.ip_forward=1
```

Save the file and then enter

```
sudo sh -c "echo 1 > /proc/sys/net/ipv4/ip_forward"
```

to activate the forwarding. You'll now have to specify the routing rules that will connect the Ethernet port (eth0) that's connected to the internet and the Wi-Fi access point (wlan0) which is exposed to the devices within your network:

```
sudo iptables -t nat -A POSTROUTING -o eth0 -j
MASQUERADE
sudo iptables -A FORWARD -i eth0 -o wlan0 -m state --state
RELATED,ESTABLISHED -j ACCEPT
sudo iptables -A FORWARD -i wlan0 -o eth0 -j ACCEPT
```

By default, these rules will be flushed when you restart the Pi. To make them permanent, first run:

```
sudo sh -c "iptables-save > /etc/iptables.ipv4.nat"
```

Then edit the ***/etc/network/interfaces*** file, scroll down to the very end and add

```
up iptables-restore < /etc/iptables.ipv4.nat
```

what this does is loads the rules when the devices are activated on boot.

Your own hostapd

Sometimes even though a wireless adaptor works out of the box on the Raspberry Pi, it might throw errors when it's asked to serve as an access point. This is especially true of cards that use Realtek chipsets, like the one we've used – MicroNext MN-WD152B – which uses the RTL8192CU chipset. While it works right off the bat for browsing the web, it doesn't work with the *hostapd* client in Raspbian's repository. It turns out Realtek has its own version of *hostapd* client which you'll have to use in case you are in the same predicament as us.

To download the file, head to Realtek's download section (**http://bit.ly/RealtekWiFiDrivers**) and select your chipset from the ones listed. This takes you to a page that lists the drivers for your chipsets. From this page grab the driver for Linux, which will download a compressed zip file with a long-windy name. In our case this was called **RTL8188C_8192C_USB_linux_v4.0.2_9000.20130911.zip**. We'll just refer to it as **driver.zip**.

Copy this file to the Raspberry Pi using *scp* using something like:

```
scp driver.zip  pi@192.168.2.100:/home/pi
```

This copies the file to the Pi's home directory. Now extract the file with

```
unzip driver.zip
```

and **cd** into the **wpa_supplicant_hostapd** directory. It'll list several compressed tarballs. Use the **tar zxvf** command to extract the file beginning with **wpa_supplicant_hostapd**. Now **cd** into the **hostapd** directory under the extract directory. This directory has a file named Makefile. Open it in a text editor and replace the

```
CFLAGS = -MMD -O2 -Wall -g
```

line towards the top of the file with

```
CFLAGS=-MMD -Os -Wall -g
```

Save the file and enter *make* to compile the *hostapd* client. It'll take quite some time and when it's complete it'll replace the *hostapd* binary in this directory.

Before using this new version, move out the old version with:

```
sudo mv /usr/sbin/hostapd /usr/sbin/hostapd.
orig
```

Then copy over the newly compiled version with the following:

```
sudo cp hostapd /usr/sbin/
```

And give it the right permissions with:

```
sudo chmod 755 /usr/sbin/hostapd
```

You should now be able to get your access point online without any issues.

Your Pi access point is now all set. To test it restart the DHCP server with

```
sudo service isc-dhcp-server restart
```

and manually enable the access point with our configuration with the following command [Read the *'Your Own Hostapd' box, p72*, if you get an unknown driver error]:

```
sudo /usr/sbin/hostapd /etc/hostapd/hostapd.conf
```

If everything goes well, the wireless access point (TorSpot) is listed in the list of available Wi-Fi hotspots. You can connect to it from another computer or a smartphone and authenticate using the password you specified in the **hostapd.conf** file. When connected, you should be able to browse the Internet normally.

Once you have tested the new access point, let's cement the settings so that they are activated as soon as the Pi boots up. Start the hostapd and DHCP services with the

```
sudo service hostapd start
```

and

```
sudo service isc-dhcp-server start
```

commands and then update the init scripts with

```
sudo update-rc.d hostapd enable
```

and

```
sudo update-rc.d isc-dhcp-server enable
```

Now restart the Pi with

```
sudo shutdown -r now
```

When the Pi is back up again, you'll be able to connect to the new access point and browse normally.

Torify access

Your Raspberry Pi is now fully functional as a wireless hotspot. However, the data is still not anonymised. So let's add *Tor* to the mix. SSH back into the Pi and install *Tor* with

```
sudo apt-get install tor
```

When it's installed, edit *Tor*'s config file **/etc/tor/torrc** and add the following at the top:

```
Log notice file /var/log/tor/notices.log
VirtualAddrNetwork 10.192.0.0/10
AutomapHostsSuffixes .onion,.exit
AutomapHostsOnResolve 1
TransPort 9040
TransListenAddress 192.168.12.1
DNSPort 53
DNSListenAddress 192.168.12.1
```

These settings inform *Tor* about the IP address of our access point and asks that it anonymises any traffic that flows over it. Next up, we'll change the routing tables so that connections via the Wi-Fi adaptor (wlan0) are routed through *Tor*. First, flush the existing redirection and NAT rules with the

```
sudo iptables -F
```

command go on to

```
sudo iptables -t nat -F
```

command. Since, we'll still want to be able to SSH into the Pi, we'll add an exception for SSH's Port 22 with:

```
sudo iptables -t nat -A PREROUTING -i wlan0 -p tcp --dport 22 -j REDIRECT --to-ports 22
```

We'll now add two rules. The first is a passthrough rule for DNS lookups and the second directs all TCP traffic to *Tor*'s port 9040:

```
sudo iptables -t nat -A PREROUTING -i wlan0 -p udp --dport 53 -j REDIRECT --to-ports 53
sudo iptables -t nat -A PREROUTING -i wlan0 -p tcp --syn -j REDIRECT --to-ports 9040
```

Like before, these rules won't be carried on to the next session. To load them on reboot, all you have to do is save them to the NAT save file like before with

```
sudo sh -c "iptables-save > /etc/iptables.ipv4.nat"
```

In the previous section, we've already configured the **/etc/network/interfaces** file to load the contents of this file when the interfaces are activated.

You can now enable the Tor service with

```
sudo service tor start
```

and update the relevant boot scripts with

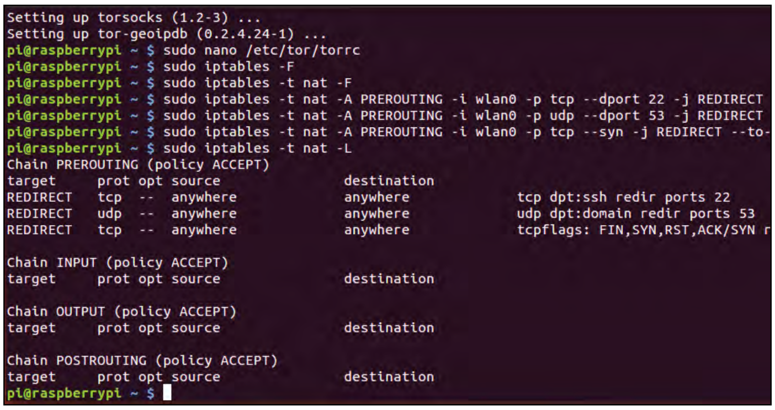
```
sudo update-rc.d tor enable.
```

That's it. Now restart the Pi. When it's back up again, you'll be able to connect to the Pi hotspot, TorSpot, as before. However, unlike as before all your traffic will now be routed through the *Tor* network.

You can verify that this is happening by heading to check **https://torproject.org** from any device that's connected to TorSpot. The page will also list your IP address which will not be that of your ISP. Visit this page from another device connected to TorSpot and it'll show a different address. Congratulations, you can now anonymously browse the web on all your devices! LXF

Quick tip

We've used Google's DNS service in this tutorial, but you can use another service like OpenDNS or your ISPs DNS servers by pointing to them in the **/etc/dhcp/dhcpd.conf** file.



```
Setting up torsocks (1.2-3) ...
Setting up tor-geoipdb (0.2.4.24-1) ...
pi@raspberrypi ~ $ sudo nano /etc/tor/torrc
pi@raspberrypi ~ $ sudo iptables -F
pi@raspberrypi ~ $ sudo iptables -t nat -F
pi@raspberrypi ~ $ sudo iptables -t nat -A PREROUTING -i wlan0 -p tcp --dport 22 -j REDIRECT
pi@raspberrypi ~ $ sudo iptables -t nat -A PREROUTING -i wlan0 -p udp --dport 53 -j REDIRECT
pi@raspberrypi ~ $ sudo iptables -t nat -A PREROUTING -i wlan0 -p tcp --syn -j REDIRECT --to-
pi@raspberrypi ~ $ sudo iptables -t nat -L
Chain PREROUTING (policy ACCEPT)
target     prot opt source               destination
REDIRECT   tcp  --  anywhere             anywhere             tcp dpt:ssh redir ports 22
REDIRECT   udp  --  anywhere             anywhere             udp dpt:domain redir ports 53
REDIRECT   tcp  --  anywhere             anywhere             tcpflags: FIN,SYN,RST,ACK/SYN r

Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain POSTROUTING (policy ACCEPT)
target     prot opt source               destination
pi@raspberrypi ~ $ 
```

› **Verify the traffic redirection rules with the sudo iptables -t nat -L command.**

Tor-in-a-box options

If you find this tutorial too cumbersome, or want to set up something for a non-technical friend or relative, there are several ready-made hardware solutions that can anonymise all their web traffic in a similar fashion.

There's the OnionPi Pack from AdaFruit (**http://bit.ly/AdaOnionPi**) which includes a Raspberry Pi B+ and a compatible USB Wi-Fi adaptor along with a case for the Pi, cables, SD card and everything else you need to setup your Torrified Wi-Fi hitspot. The bundle costs $80. However, you'll still have to follow the instructions and set it yourself.

If you'd rather have something more plug and play, there's the SafePlug from the guys who bought us PogoPlug. It's a $49 device that plugs into your wireless router and once activated routes all traffic over the Tor network. A neater and smaller alternative is the Anonabox (**www.anonabox.com**). It initially launched on Kickstarter but after its funding was suspended it relaunched on Indiegogo. Here it was listed at $51 and surpassed its funding target in early January 2015 and will begin shipping in February 2015. Anonabox is a router that you can directly connect to via Wi-Fi or Ethernet.

Another router-based option is Portal which stands for Personal Onion Router To Assure Liberty. The project produces a pre-built software image for several TP-Link routers. You can simply flash the Portal firmware image onto these router following the instructions on the project's website (**https://github.com/grugq/portal**).

Encryption Set up encryptfs and tweak it for auto-mounting and easy access

Encryption: full drive protection

Neil Bothwick shows you how to keep your files safe from prying eyes, even other users of your computer, using ecryptfs.

Our expert

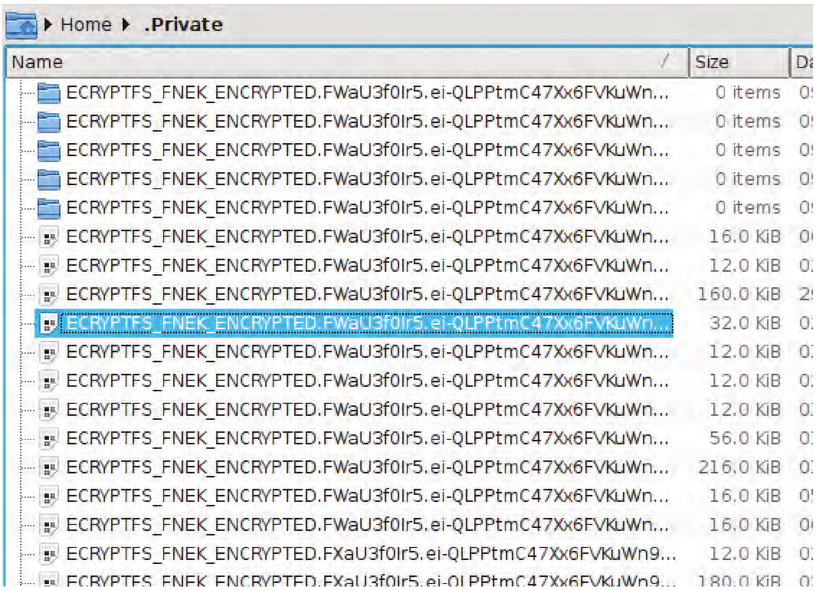
Neil Bothwick has a great deal of experience with booting up, as he has a computer in every room, but not as much with rebooting since he made the switch from Windows to Linux.

Last year, when everyone was interested in privacy in the aftermath of Edward Snowden's revelations, we looked at using *cryptsetup* to encrypt whole disk partitions with the Linux kernel's *dm-crypt* facilities, but there are other encryption systems available. There are several ways of encrypting data on your computer.

The method we looked at before encrypted a whole block device, usually a disk partition. This is good for whole system encryption, but makes everything available once the system is booted. There was also *TrueCrypt*, which works with either whole devices or virtual disks (a large file that acts like a disk). We looked at using *TrueCrypt* back in **LXF161**,[*Tutorials, p84*] but it was abandoned in 2014, and although there have been a couple of forks many people are still using the 7.1a version (the final, neutered 7.2 version only allows viewing of *TrueCrypt* volumes). Another alternative is for the filesystem to handle the encryption, as ZFS does on Sun systems, but none of the main Linux filesystems provide encryption themselves.

Introducing ecryptfs

The next option, and the one we are concerned with today, is what is called a stacked filesystem, where you mount one filesystem on top of another, and this is what *ecryptfs* uses (*cryptsetup*, which we've covered before uses stacked block devices, below the filesystem).



Home ▸ .Private

Name	Size	Da
ECRYPTFS_FNEK_ENCRYPTED.FWaU3f0Ir5.ei-QLPPtmC47Xx6FVKuWn...	0 items	0
ECRYPTFS_FNEK_ENCRYPTED.FWaU3f0Ir5.ei-QLPPtmC47Xx6FVKuWn...	0 items	0
ECRYPTFS_FNEK_ENCRYPTED.FWaU3f0Ir5.ei-QLPPtmC47Xx6FVKuWn...	0 items	0
ECRYPTFS_FNEK_ENCRYPTED.FWaU3f0Ir5.ei-QLPPtmC47Xx6FVKuWn...	0 items	0
ECRYPTFS_FNEK_ENCRYPTED.FWaU3f0Ir5.ei-QLPPtmC47Xx6FVKuWn...	0 items	0
ECRYPTFS_FNEK_ENCRYPTED.FWaU3f0Ir5.ei-QLPPtmC47Xx6FVKuWn...	16.0 KiB	0
ECRYPTFS_FNEK_ENCRYPTED.FWaU3f0Ir5.ei-QLPPtmC47Xx6FVKuWn...	12.0 KiB	0
ECRYPTFS_FNEK_ENCRYPTED.FWaU3f0Ir5.ei-QLPPtmC47Xx6FVKuWn...	160.0 KiB	2
ECRYPTFS_FNEK_ENCRYPTED.FWaU3f0Ir5.ei-QLPPtmC47Xx6FVKuWn...	32.0 KiB	0
ECRYPTFS_FNEK_ENCRYPTED.FWaU3f0Ir5.ei-QLPPtmC47Xx6FVKuWn...	12.0 KiB	0
ECRYPTFS_FNEK_ENCRYPTED.FWaU3f0Ir5.ei-QLPPtmC47Xx6FVKuWn...	12.0 KiB	0
ECRYPTFS_FNEK_ENCRYPTED.FWaU3f0Ir5.ei-QLPPtmC47Xx6FVKuWn...	12.0 KiB	0
ECRYPTFS_FNEK_ENCRYPTED.FWaU3f0Ir5.ei-QLPPtmC47Xx6FVKuWn...	56.0 KiB	0
ECRYPTFS_FNEK_ENCRYPTED.FWaU3f0Ir5.ei-QLPPtmC47Xx6FVKuWn...	216.0 KiB	0
ECRYPTFS_FNEK_ENCRYPTED.FWaU3f0Ir5.ei-QLPPtmC47Xx6FVKuWn...	16.0 KiB	0
ECRYPTFS_FNEK_ENCRYPTED.FWaU3f0Ir5.ei-QLPPtmC47Xx6FVKuWn...	16.0 KiB	0
ECRYPTFS_FNEK_ENCRYPTED.FXaU3f0Ir5.ei-QLPPtmC47Xx6FVKuWn9...	12.0 KiB	0
ECRYPTFS_FNEK_ENCRYPTED.FXaU3f0Ir5.ei-QLPPtmC47Xx6FVKuWn9...	180.0 KiB	0

› This is how your files look after encryption, and their contents are equally unintelligible

Because *ecryptfs* works on top of the normal filesystem, it's not restricted to entire disk partitions, it can be used to encrypt individual directories. This is the method Ubuntu uses to provide encrypted home directories if you choose that option during installation. It is easiest to explain with an example. The *ecryptfs* filesystem itself is contained in the Linux kernel, but you will need to install the **ecryptfs-utils** package for the tools to work with it. Create two directories called **crypt** and **plain**, then you can create an encrypted directory with this command:

```
sudo mount.ecryptfs crypt plain
```

You will be asked a number of questions, obviously you should choose a password that is both secure and memorable (or store it somewhere safe). Most of the rest can be left as the defaults with the possible exception of Enable Filename Encryption that you may want to set to yes. Now copy some files to plain then look in **crypt**. You will see the same filenames if you didn't enable filename encryption, otherwise you will see encrypted names. Either way, the contents will be encrypted; try viewing one of the files. Now unmount it with:

```
sudo umount plain
```

The readable versions of the files have disappeared, leaving only the encrypted versions. Run the above **mount** command and the contents of **plain** will reappear. This method of mounting is cumbersome but it illustrates how *ecryptfs* functions. The filesystem you mounted on **plain** is virtual, it exists only in memory, the only data written to disk are the encrypted files in **crypt**. Once you unmount the **plain** version your data is protected, and cannot be read again until you mount it, which requires your password.

Convenient encryption

There is, of course, a more convenient way of setting up an encrypted directory for a user that doesn't require **sudo** or answering questions - run this as your normal user:

```
ecryptfs-setup-private
```

The command will ask for your login password and then a passphrase for the encrypted directory. The former is used to lock the latter, which you can leave that blank and have *ecryptfs* generate a secure passphrase automatically. This creates three directories: **.Private** contains your encrypted data, **Private** is the mountpoint for the decrypted contents and **.ecryptfs** contains files that are used to mount your directory. As the passphrase itself is encrypted, you should make a copy and store it somewhere secure, such as a USB

Pros and cons of ecryptfs

Ecryptfs has a number of advantages over *LUKS/dm-crypt*:

» **Back up to cloud** As the encryption is at file level, you can backup your **.Private** directory to a cloud service or external drive without worrying about your data being accessible to others. Just make sure you backup **.cryptfs** and your passphrase some separate and secure.

» **Multi-user security** *Ecryptfs* can encrypt directories separately for each user.

» **Directory** *Ecryptfs* can also be used on system directories and swap, with a suitable fstab entry, but it will prompt for a passphrase.

» **Login to read** A user's data is only available when the user is logged in, and even then *ecryptfs* defaults to making it only readable by that user (and root, of course).

There are, however, some disadvantages too:

» **Many files** It is slower dealing with directories containing many files, although this can be mitigated (at the expense of security) by having *ecryptfs* not encrypt filenames.

» **Large files** Because each file is encrypted separately, the files all increase in size, which can be significant with a large number of small files, like an email or browser cache.

» **Not cross-platform** *Ecryptfs* is Linux only, using features of the kernel, which won't be a problem for everyone. As far as we are aware, there's no reliable way to read Windows files.

key nowhere near your computer:

```
ecryptfs-unwrap-passphrase ~/.ecryptfs/wrapped-passphrase
>/somewhere/safe/ecryptfs_passphrase
```

Now you can mount and unmount your private data with these commands, or use the desktop icon it provides.

```
ecryptfs-mount-private
ecryptfs-umount-private
```

This creates a single, encrypted directory in your **home**, but what if you want more? Let's say you want your Documents and Accounts directories encrypted but see no point in encrypting Photos or Music (why waste time decrypting large files that hold nothing private). The easy answer is to move the directories into **Private** and create symbolic links back to their original locations, like this:

```
mv Documents Private
ln -s Private/Documents Documents
```

Make sure **Private** is mounted when you do this, then your files will only be available when the *ecryptfs* filesystem is mounted, otherwise it will just show up as a broken link.

Automatic mounting

You give your login password to unlock the *ecryptfs* passphrase to mount the filesystem (you can use the -w option to **ecryptfs-setup-private** if you want to use an independent password) so you may be asking why when you've already just given a password to login, you need to give it again to mount your private files? This is a valid question, if you know it once, I'm sure you can remember it again a few seconds later. If you prefer, you can have your **Private** directory automatically mounted when you login (and unmounted when you logout), thanks to the magic of PAM. As root, insert this line into **/etc/pam.d/common-auth**:

```
auth    required    pam_ecryptfs.so unwrap
```

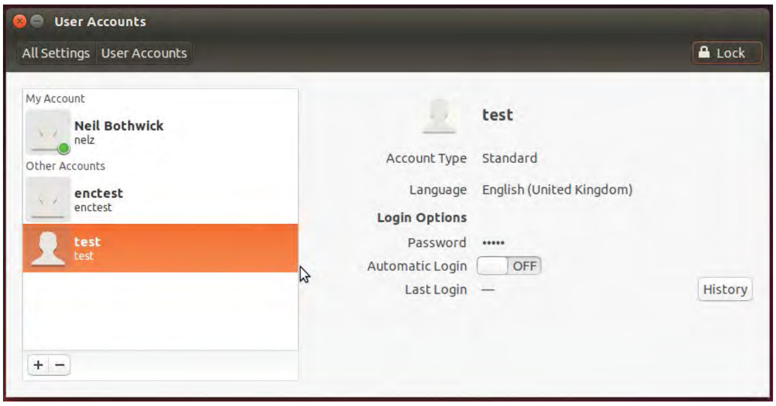
and this one into **/etc/pam.d/common-session**:

```
session optional    pam_ecryptfs.so unwrap
```

Now PAM will mount your *ecryptfs* **home** directory when you login. This will not happen if you have auto-login enabled, otherwise you would have no security at all.

Encrypted $HOME

If all of this looks a little familiar, that is probably because you have used the encrypted **home** directory feature in Ubuntu, which also uses *ecryptfs*. But this a standard kernel feature not restricted to one distro (ChromeOS also uses *ecryptfs* behind the scenes). Ubuntu doesn't just set up a **Private** directory when you install it, but it encrypts your entire **home** directory. So the simplest way to get a fully encrypted **home** directory may seem to be to install Ubuntu and choose that option. There are a couple of reasons you may not want to do this: you may use a different distro or you may already use Ubuntu but don't want to start again with a new installation.



My Account

Neil Bothwick
nelz

Other Accounts

enctest
enctest

test
test

test

Account Type Standard

Language English (United Kingdom)

Login Options

Password •••••

Automatic Login OFF

Last Login —

There's a single command that will convert your entire **home** directory to *ecryptfs*, but there are a couple of caveats. You must have no files in use in the **home** directory, which means that the user mustn't be logged in, and you need free space of up to 2.5 times the current size of your **home** directory for the conversion process (mainly because encrypted and unencrypted copies of your files are stored until the job is done). So log out and log in as another user with admin rights then run:

```
sudo ecryptfs-migrate-home --user <yourusername>
```

After the process completes, you must log in as that user before rebooting, to complete the setup and make sure everything is working. Once that is done and you have verified that your files are there and readable, you can delete the original unencrypted files that are still in **/home/user.some_random_string**. Be aware that deleting that directory does not remove all of your unencrypted data from your hard drive, only the directory table. To be fully secure, you should overwrite all unused space with random data.

```
dd if=/dev/urandom of=somefile bs=4k
rm somefile
```

This creates a file of random data that fills the drive and then deletes it to return the space to you.

Whether you use **ecryptfs-setup-private** or **ecryptfs-migrate-home**, you should use **ecryptfs-unwrap-passphrase** to save the passphrase to a safe place. If you don't keep a copy of your passphrase, you won't be able to access your data if the **.ecryptfs** directory is lost or damaged. LXF

› If your distro does not permit root login, like Ubuntu, create a spare user with admin rights when encrypting your home directory.

Power management Using suspend, hibernate and Wake on LAN

Power: States and governors

Jonni Bidwell dabbles with the many facades of managing power in Linux.

Our expert

Jonni Bidwell is – mostly – still in his hibernatory winter state. The motherboard of the cursed Linux box abandoned unto him at **LXF** Towers is resolute in its ACPI-uncompliance.

At one stage, power management on Linux was regarded as a bit of a joke. And we're not talking quirks with relatively advanced features such as hibernate and suspend – oh no, it used to be that a simple **shutdown -h now** could, rather than gracefully power off the system, send it into some nightmarish limbo state, whence one's only recourse was to hard reset the machine. Upon starting up the unfortunate machine, one might have to endure a lengthy *fsck* (we're talking back in the pre-journal days), which may unearth corrupted data. Worse still, hard drives could end up damaged following the whole ordeal. While these days are, for the most part behind us, many people still run into other difficulties with power management.

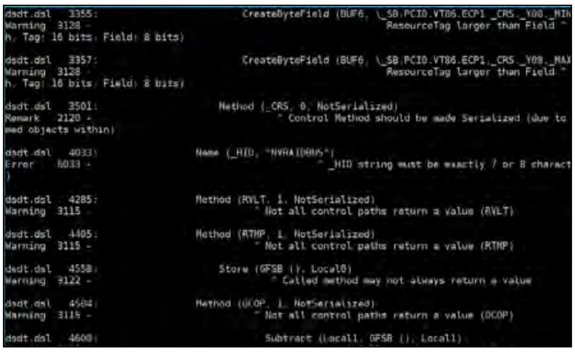
There are open standards which govern power management (also hardware discovery), namely the ACPI (Advanced Configuration and Power Interface) BIOS standard developed by Intel, Toshiba and Microsoft, and first released in 1996. ACPI replaced the old and no longer fit for purpose APM (advanced power management), which provided a rudimentary bridge between BIOS and the OS. An ACPI-aware OS is, among other things, able to react to button or lid events, triggering shutdown or standby states. Thus Windows 98, the first such OS, could do away with the 'It's now safe to turn off your computer' shutdown screen of its predecessor.

Unfortunately, motherboard manufacturers were not particularly adept at adhering to these standards, which was largely fine if people used the protocol-deviant Windows drivers provided with their motherboards, but it led to a world of pain for users of other OSes. Besides compliance issues, ACPI itself has come under fire, Linus in particular decried it as "a complete design disaster in every way" in 2003 (as is his way, he further advised any Intel employee having a hand in it to "shoot yourself now, before you reproduce"). His objections stemmed from, besides it being a generally messy and overly complex system, the way that it requires AML bytecode to be run unchecked by the kernel. The ASL (ACPI Source Language) code which gives rise to this bytecode may not be available, which makes debugging a pain.

Buggy BIOSes

The 2.6 series kernels heralded a new era of ACPI support, early (pre-2001), buggy implementations were blocked, and, in theory, a typical Linux PC circa 2004 could understand the six Power Sleep states S0-S5, Device states, Processor and Performance states. This was particularly helpful on laptops (which had recently become affordable and portable), whose users were able to at once conserve battery life and obviate lengthy boot times, courtesy of Suspend to RAM/Disk. New processors brought ACPI innovations into the server room too by throttling or powering down idle machines, temperatures and bills were lowered.

Unfortunately buggy BIOSes persist, particularly on machines older than five years, so if you do run into issues, it's worth checking if an updated BIOS is available from your motherboard manufacturer. If this doesn't solve your problem and you feel like an adventure, then read up on repairing buggy DSDT tables. The Arch Linux wiki page is a good place



```
dsdt.dsl   3355:                     CreateByteField (BUF6, \_SB.PCI0.VT86.ECP1._CRS._Y08._MIN
Warning  3128 -                                                     ResourceTag larger than Field ^
h, Tag: 16 bits, Field: 8 bits)

dsdt.dsl   3357:                     CreateByteField (BUF6, \_SB.PCI0.VT86.ECP1._CRS._Y08._MAX
Warning  3128 -                                                     ResourceTag larger than Field ^
h, Tag: 16 bits, Field: 8 bits)

dsdt.dsl   3501:                  Method (_CRS, 0, NotSerialized)
Remark   2120 -                          ^ Control Method should be made Serialized (due to
med objects within)

dsdt.dsl   4033:               Name (_HID, "NVRAIDBUS")
Error    6033 -                                   ^ _HID string must be exactly 7 or 8 charact
)

dsdt.dsl   4285:               Method (RVLT, 1, NotSerialized)
Warning  3115 -                        ^ Not all control paths return a value (RVLT)

dsdt.dsl   4405:               Method (RTMP, 1, NotSerialized)
Warning  3115 -                        ^ Not all control paths return a value (RTMP)

dsdt.dsl   4558:                  Store (GFSB (), Local0)
Warning  3122 -                          ^ Called method may not always return a value

dsdt.dsl   4584:               Method (UCOP, 1, NotSerialized)
Warning  3115 -                        ^ Not all control paths return a value (UCOP)

dsdt.dsl   4600:                     Subtract (Local1, GFSB (), Local1)
```

› **This is what a broken DSDT looks like, the spec is rather fussy about the length of identifiers.**

Ice ice, baby

Enhanced suspend and hibernate functionality is provided by the TuxOnIce kernel patchset. This allows, among other things, more control over the hibernate image: such as where it's stored, compression and the ability to encrypt it.

If you use the Ubuntu distribution then you can add the TuxOnIce PPA and install the kernel like this:

```
$ sudo add-apt-repository ppa:tuxonice/ppa
$ sudo apt-get update
$ sudo apt-get install tuxonice-userui linux-generic-tuxonice linux-headers-generic-tuxonice
```

Arch Linux users can use the **linux-ice** package from the AUR, others will want to grab the patch from **www.tuxonice.net** and follow their distribution's instructions for compiling custom kernels. On non-Ubuntu systems, some additional setup is required, and this is very much distro-dependent. This is mostly concerned with tweaks to the *initramfs/initrd* – the initial harness which *Grub* loads for the kernel. It needs to have resume support as well load the **lzo** module so that decompression can happen.

TuxOnIce allows for nice progress bars to be displayed through the framebuffer layer during hibernate and resume. This is achieved through the **fbsplash** and **tuxonice-uerui** packages. It also allows you to interrupt the suspend/resume process or force a reboot. The TuxOnIce team also maintains **hibernate-script** a wrapper which provides easy access to both its own and *uswsusp*'s hibernate back-ends. This means you can use the hybrid Suspend to Both state, which resumes quickly from RAM if the battery survived the break or slowly from disk otherwise.

to start **https://wiki.archlinux.org/index.php/DSDT**, but since the table needs to be embedded into your kernel image, you'll need to be au fait with kernel compilation. The kernel's documentation provides some good advice for debugging hibernate and suspend issues, too (see **http://bit.ly/BasicPMDebug**).

All ACPI functionality used to be controlled by the acpid daemon, but much of it is now provided by either your desktop environment or *Systemd*. For the latter, you can specify what happens when you close your laptop's lid, push the power button etc, by editing the file **/etc/systemd/logind.conf**. The relevant options and actions are largely self-explanatory:

```
HandlePowerKey=poweroff
HandleSuspendKey=suspend
HandleHibernateKey=hibernate
HandleLidSwitch=suspend
HandleLidSwitchDocked=ignore
```

Of course, if you want to test things the grown-up way then you'll need some command-line fu. In particular, you can trigger power changes by writing directly to the **/sys/power/state** interface. For example, to enter the suspend (S3, STR) state, issue the following (as root):

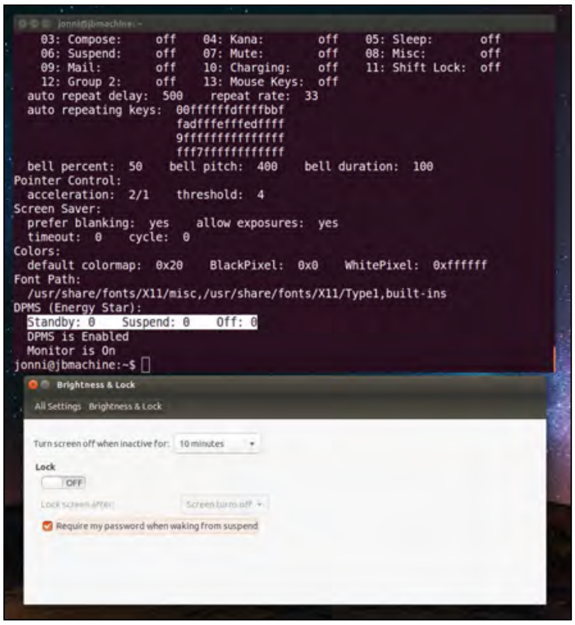
```
# echo mem > /sys/power/state
```

Besides **mem**, you can also enter the slightly more power-hungry but faster-resuming **freeze** (S1) or **standby** (S2) states. Use **disk** to enter the hibernate (S4) state. For this to work you'll need to have a sufficiently large **swap** partition, since your RAM contents are going to get dumped here. They will be compressed, so even if your **swap** partition is smaller than your quantity of RAM it's still possible for this to work. The machine will **shutdown** (S5) once the hibernate image is written, so in order for it to resume we need to tell the kernel where this is located. This requires adding an option such as

```
resume=/dev/sda2
```

as a kernel parameter, where **sda2** is your swap partition. You can use the more robust (also more lengthy) UUID of said partition here, if you prefer. To effect the changes, as root add the desired option to **GRUB_CMDLINE_LINUX** in **/etc/default/grub** and then run **grub-update**.

It is worth looking at **pm-utils** which is a collection of wrappers around the kernel's (or TuxOnIce's or uswsusp's) powerdown machinations. It provides workarounds for various motherboard quirks, as well as the ability to unload troublesome modules prior to suspending. A word of warning though: If a kernel update is applied, and then you hibernate your machine, and then resume it, then the new kernel will get very upset with the resume image. So don't hibernate following kernel updates.



```
jonni@jbmachine: ~
   03: Compose:     off    04: Kana:        off    05: Sleep:       off
   06: Suspend:     off    07: Mute:        off    08: Misc:        off
   09: Mail:        off    10: Charging:    off    11: Shift Lock:  off
   12: Group 2:     off    13: Mouse Keys:  off
  auto repeat delay:  500    repeat rate:  33
  auto repeating keys:  00ffffffdffffbbf
                        fadfffefffedffff
                        9fffffffffffffff
                        fff7ffffffffffff
  bell percent:  50    bell pitch:  400    bell duration:  100
Pointer Control:
  acceleration:  2/1    threshold:  4
Screen Saver:
  prefer blanking:  yes    allow exposures:  yes
  timeout:  0    cycle:  0
Colors:
  default colormap:  0x20    BlackPixel:  0x0    WhitePixel:  0xffffff
Font Path:
  /usr/share/fonts/X11/misc,/usr/share/fonts/X11/Type1,built-ins
DPMS (Energy Star):
  Standby: 0    Suspend: 0    Off: 0
  DPMS is Enabled
  Monitor is On
jonni@jbmachine:~$
```

Turn screen off when inactive for: 10 minutes

Lock: OFF

Lock screen after: Screen turns off

☑ Require my password when waking from suspend

› **Sometimes the DPMS settings shown in xset will be clobbered by your desktop. Here, Unity has taken responsibility for turning off the screen.**

Who's the governor?

Modern processors (and even 12-year-old Athlon XPs if you have an nForce 2 motherboard and some time on your hands) all support some degree of frequency stepping. This is enabled (usually by default) in the BIOS. Intel's incarnation of this is dubbed Enhanced Speedstep and AMD's goes by Cool and Quiet, or PowerNow. The idea is that, when feasible, the processor is slowed down and core voltages dropped (in some cases cores are powered off entirely). This means that system temperatures will drop, and, in turn, that fans can be slowed down and energy bills significantly reduced. Control of this mechanism is done through the kernel's CPUFreq subsystem, and is generally set up by default. Different profiles, which are called governors, are available to suit various situations. You can check which one is active on your system with:

```
$ cat /sys/devices/system/cpu/cpu0/cpufreq/scaling_governor
```

»

Quick tip

If you run into difficulties (eg no graphics after the system is resumed from RAM), check out the **uswsusp** package which allows further customisation of the suspension process. It allows you to tweak graphics card settings pre- and post-resume, as well as providing support for encrypted hibernation images.

» For a desktop system the default **ondemand** governor is generally the best choice – it will keep the frequency at its lowest until the CPU is stressed, then it will be raised as high as necessary, until the panic is over.

The governor is very responsive, capable of changing the frequency hundreds of times per second, so you won't notice any latency when more megahertz are required. However, you might notice some slowdown if you have a particularly variable workload, heavy compilation jobs for example, where the bottleneck oscillates chaotically between CPU and disk I/O operations . The other governors are: **performance** (keeps the CPU at its maximum frequency), **conservative** (switches frequency gradually, higher latency than ondemand), and **powersave** (lowest frequency). The latter might be something of a misnomer under heavy workloads though, since it would be more energy efficient to process these faster, rather than labour the CPU. The active governor can be changed by writing to the ***/sys*** entry above as root, for instance:

```
$ sudo -i
# echo performance > /sys/devices/system/cpu/cpu0/cpufreq/
scaling_governor
```

Besides these, you can also set the scaling governor to **userspace**, which isn't really a governor in itself, but rather indicates that a userspace program will manipulate frequencies. Such a program is *cpupower*, which (replacing the deprecated *cpufreqd*) allows you to set custom minimum and maximum frequencies. In general, it's not necessary to install this, but you may want to do so nonetheless.
For example, on old laptops it's common that too onerous a workload will cause overheating and the BIOS to lock the CPU into a cripplingly low-powered state. In this case it's convenient to preclude the CPU reaching its maximum frequency, which could be achieved with something like:

```
# cpupower frequency-set -u 1600MHz
```

You can examine the frequencies that are supported by your CPU with:

```
$ cpupower -c 0 frequency-info
```

where the **-c** option determines a particular CPU core. It's possible to set frequencies on a per-core basis too. In the event that this command returns unexpected results, eg that your CPU can only operate very much slower than its marketed frequency, then you'll probably want to investigate if a BIOS upgrade is available, particularly on older hardware. As mentioned earlier, manufacturers sometimes aren't very good at adhering to standards, which in this case results in the BIOS reporting inaccurate frequencies. If no update is available and you're happy to have your CPU running at full speed all the time, then you can always disable Speedstep *et al* in the BIOS.

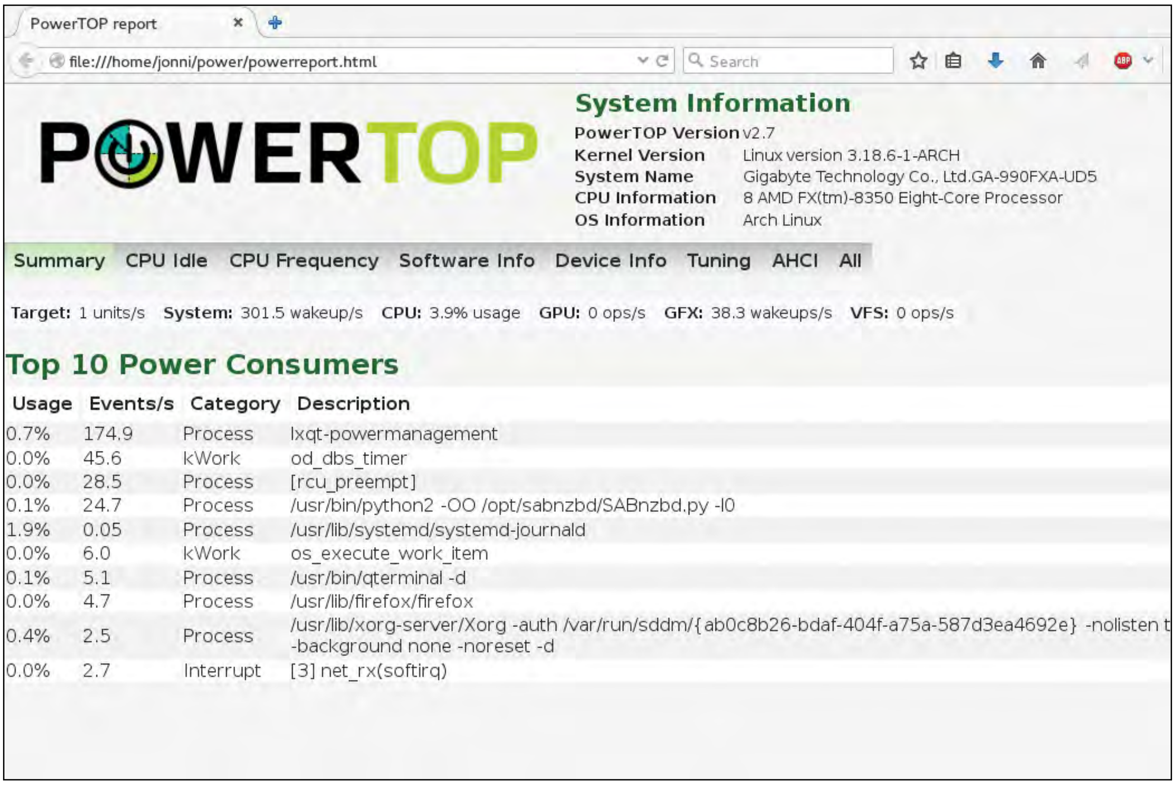
If you install Intel's *powertop* program then you can see how much time has been spent in each frequency. *Powertop* also shows which programs are generating the most wake-ups, and can even tune various and diverse power-management settings through the ***/sys*** interface. Running

```
# powertop –html=powerreport.html
```

will generate a report showing this information.

Wake 'n' bake

The system can be re-animated from the suspend or hibernate states by keyboard or mouse activity. With well-behaved hardware though, one can also wake a remote machine by sending a so-called magic packet via the LAN. Note: this can work with specific wireless adaptors, but it's a hit and miss affair, so we're concentrating only on wired WoL here. The first thing to check is that WoL is enabled in the power settings of



System Information

PowerTOP Version v2.7
Kernel Version Linux version 3.18.6-1-ARCH
System Name Gigabyte Technology Co., Ltd.GA-990FXA-UD5
CPU Information 8 AMD FX(tm)-8350 Eight-Core Processor
OS Information Arch Linux

Target: 1 units/s System: 301.5 wakeup/s CPU: 3.9% usage GPU: 0 ops/s GFX: 38.3 wakeups/s VFS: 0 ops/s

Top 10 Power Consumers

Usage	Events/s	Category	Description
0.7%	174.9	Process	lxqt-powermanagement
0.0%	45.6	kWork	od_dbs_timer
0.0%	28.5	Process	[rcu_preempt]
0.1%	24.7	Process	/usr/bin/python2 -OO /opt/sabnzbd/SABnzbd.py -l0
1.9%	0.05	Process	/usr/lib/systemd/systemd-journald
0.0%	6.0	kWork	os_execute_work_item
0.1%	5.1	Process	/usr/bin/qterminal -d
0.0%	4.7	Process	/usr/lib/firefox/firefox
0.4%	2.5	Process	/usr/lib/xorg-server/Xorg -auth /var/run/sddm/{ab0c8b26-bdaf-404f-a75a-587d3ea4692e} -nolisten t -background none -noreset -d
0.0%	2.7	Interrupt	[3] net_rx(softirq)

› **The idlewatcher component of LXQt's power management seems to be something of an attention seeker. Oh, the irony.**

your BIOS. Then install **ethtool** and check that your adaptor (call it eth0) supports WoL:

```
$ ethtool eth0 | grep Wake
Supports Wake-on: pumbg
Wake-on: b
```

The first line shows the type of packets which the adaptor can respond, in this case: PHY, Unicast, Multicast, Broadcast and maGic packet activity. For WoL to work, we need this to be set to the latter. If this is not the case, make it so with:

```
# ethtool -s eth0 wol g
```

This setting won't survive a reboot, so if you needed it permanently you'll want to add this command to your startup somewhere, either as a post-up networking script or through a udev rule. For the udev route, create a file **/etc/udev/rules.d/50-wol.rules** with the contents:

```
ACTION=="add", SUBSYSTEM=="net", KERNEL=="eth0",
RUN+="/usr/bin/ethtool -s %k wol g"
```

You'll need the MAC address of the target machine, since magic packets operate below the IP layer. This is a 12 hexdigit identifier which you can easily find out from:

```
# ip link
```

Waking up the target machine will require a WoL application to be installed on the host machine. One such tool is provided by the **wol** package and you can also find tools for Android and various other operating systems

Having installed **wol** and got everything set up then hopefully you may awaken a slumbering machine with

```
# wol aa:bb:cc:dd:ee:ff
```

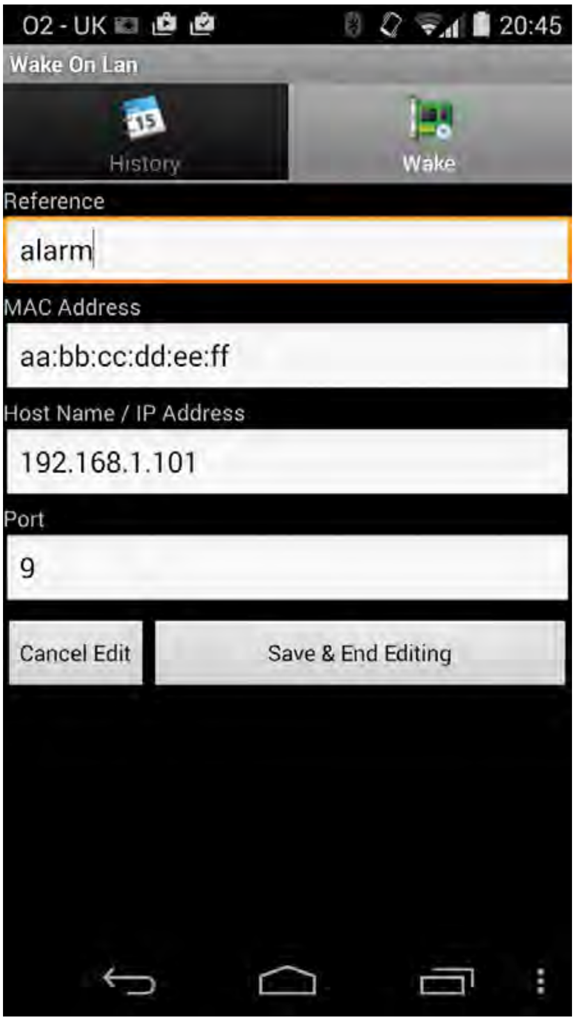
substituting for your target machine's MAC address. It's good practice to specify the machine's hostname or IP address with the **-i** option.

ARP magic

Magic packets can, in theory, be sent across the internet too – they are usually sent as UDP datagrams to port 9, some routers, however, will have difficulty dealing with them. Forwarding traffic on UDP port 9 to the target machine's IP will not suffice, since our machine will not be stirred by these high-level noises. Instead, the router must be configured to forward this traffic to the broadcast address of the internal network (eg **192.168.1.255**), and many devices don't permit this. It's possible to work around this if you have shell access to your router, for example if it's running DD-WRT or Tomato or something similar. The trick is to add an ARP entry with an unassigned IP address to the relevant interface, with the reserved broadcast MAC address FF:FF:FF:FF:FF:FF like this:

```
# arp -i br0 -s 192.168.1.254 FF:FF:FF:FF:FF:FF
```

and forward WoL traffic to this imaginary host.



Reference: alarm
MAC Address: aa:bb:cc:dd:ee:ff
Host Name / IP Address: 192.168.1.101
Port: 9

› There are many Wake-on-LAN apps for Android, but getting magic packets from the outside world into your home network is non-trivial.

If you are able to set this up, then the machine may be switched on remotely using the external IP or hostname assigned to your router (dynamic DNS services, such as duckdns.org are useful here) and the target machine's MAC address:

```
$ wol -p 9 -i EXTERNAL_IP_OR_HOSTNAME
aa:bb:cc:dd:ee:ff
```

And that concludes our tour of power management. It was once a fragmented system, plagued by buggy hardware, and in many ways it still is. But it mostly works nowadays. Just not on **Jonni's** computer. LXF

Quick tip

Owners of modern Intel CPUs (those bearing the Core i3, i5 or i7 designations) will probably want to check out the *i7z* program which allows manipulation of the advanced P-states afforded by these chips. Mmm, chips.

DPMS explained

All this fancy watt-saving is all well and good, but sometimes you need your machine, but not your monitor to stay on. The voodoo that controls the display's alertness is called DPMS (Display Power Management Signalling), the voodoo that makes the screen blank right before you want to use it, is one of life's great mysteries. In the majority of cases, some kind of screen blanking will work out of the box (even without an X server running), but it may not be just to your liking. There are three gradations of sleep: Suspend, Standby, Off in order of decreasing power usage or increasing resume time. Some monitors don't recognise the difference between the first two.

If you're using a fully-featured desktop environment (or even a minimal one like the newly-released LXQt 0.9,(*pictured, p78*), then you'll be able to manipulate these settings graphically, but otherwise you can change them by adding a file, say **/etc/X11/xorg.conf.d/12-dpms.conf** having the form:

```
Section "ServerLayout"
    Identifier "ServerLayout0"
    Option "StandbyTime" "45"
    Option "SuspendTime" "55"
    Option "OffTime" "60"
EndSection
```

Where the timeouts are specified in minutes. You can verify your settings by running **xset q**, if they're not what you expect then it's possible that a screensaver is overwriting your settings. The multi-talented *xset* can deal to that with a simple **xset s off**. You can also use it to experiment with the different sleep states, eg:

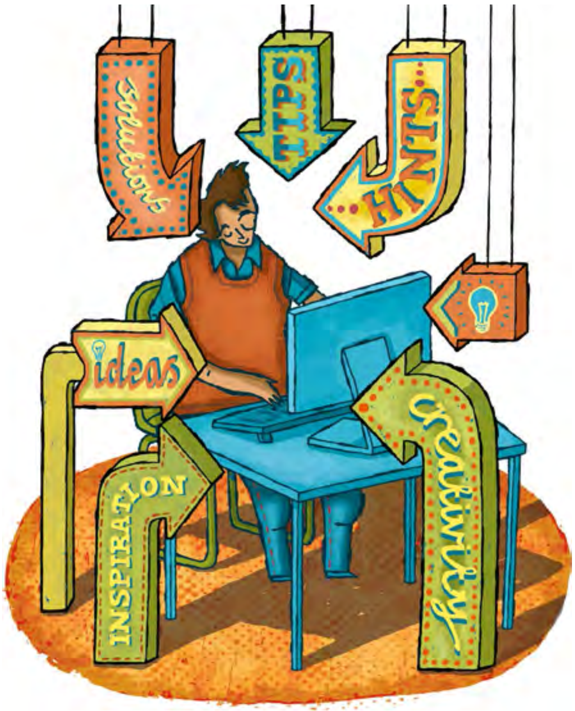
```
$ xset dpms force standby
```

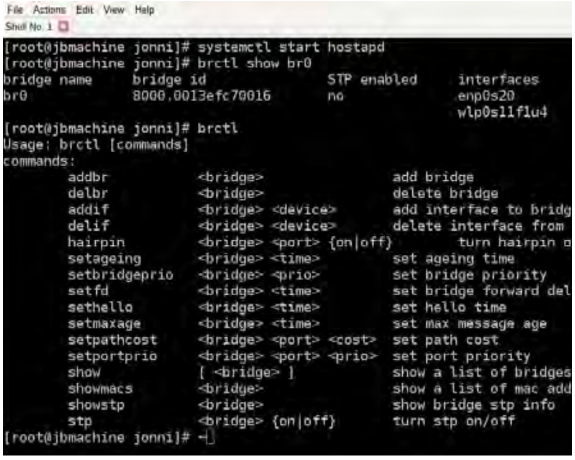
Router: set up a gateway

Jonni Bidwell throws down some iptables rules and, receiving no reply to his concerns about the temperature in the office, makes his own hotspot.

Our expert

Jonni Bidwell is shocked to learn that a computer in Leicester has control over the temperature at **LXF** towers. Today the temperature, tomorrow the minds of the content and marketing team.





```
[root@jbmachine jonni]# systemctl start hostapd
[root@jbmachine jonni]# brctl show br0
bridge name     bridge id               STP enabled     interfaces
br0             8000.0013efc70016       no              enp0s20
                                                        wlp0s11f1u4
[root@jbmachine jonni]# brctl
Usage: brctl [commands]
commands:
        addbr           <bridge>                add bridge
        delbr           <bridge>                delete bridge
        addif           <bridge> <device>       add interface to bridg
        delif           <bridge> <device>       delete interface from
        hairpin         <bridge> <port> {on|off}        turn hairpin o
        setageing       <bridge> <time>         set ageing time
        setbridgeprio   <bridge> <prio>         set bridge priority
        setfd           <bridge> <time>         set bridge forward del
        sethello        <bridge> <time>         set hello time
        setmaxage       <bridge> <time>         set max message age
        setpathcost     <bridge> <port> <cost>  set path cost
        setportprio     <bridge> <port> <prio>  set port priority
        show            [ <bridge> ]            show a list of bridges
        showmacs        <bridge>                show a list of mac add
        showstp         <bridge>                show bridge stp info
        stp             <bridge> {on|off}       turn stp on/off
[root@jbmachine jonni]# -
```

› A bridge too far. We're not meant to mess with Future Towers' networks, but brctl makes it easy.

You probably have at least one router in your house. They perform the not insignificant task of routing data from one network, eg the internet, to another, such as your home wired/wireless network. While in theory you could build on this tutorial and replace the box your ISP gave you, it's probably not the best idea – getting a Linux box with a DSL or ADSL modem in it to talk to your ISP can be tricky. Besides, many of these hubs already run Linux, and many more allow you to install a DD-WRT or OpenWRT image, both of which handle much of the ugliness for you. So why this article then? There's lots of other routing that useful!

Suppose your wireless card breaks, or needs some new firmware, leaving you without connection or a long enough patch cable to reach the router. Certainly you could replace the broken hardware or download the files from another machine, but that's hardly cricket. Instead, why not have a handy, portable, lifesaving gateway machine around, that can, via a simple crossover cable, feed the starving machine the network it so desires? You could even connect several machines this way via a simple switch – only the gateway machine needs to be able to see the internet, or whatever external network you want to share access to. You may also just be paranoid and want to install some additional firewalling for a small internal network: you could only allow certain traffic, or force all traffic to be routed via a VPN. You can even install *Wireshark* [*see* **LXF191**] on the gateway machine and check out how much Internet chatter comes out of the devices connected to it.

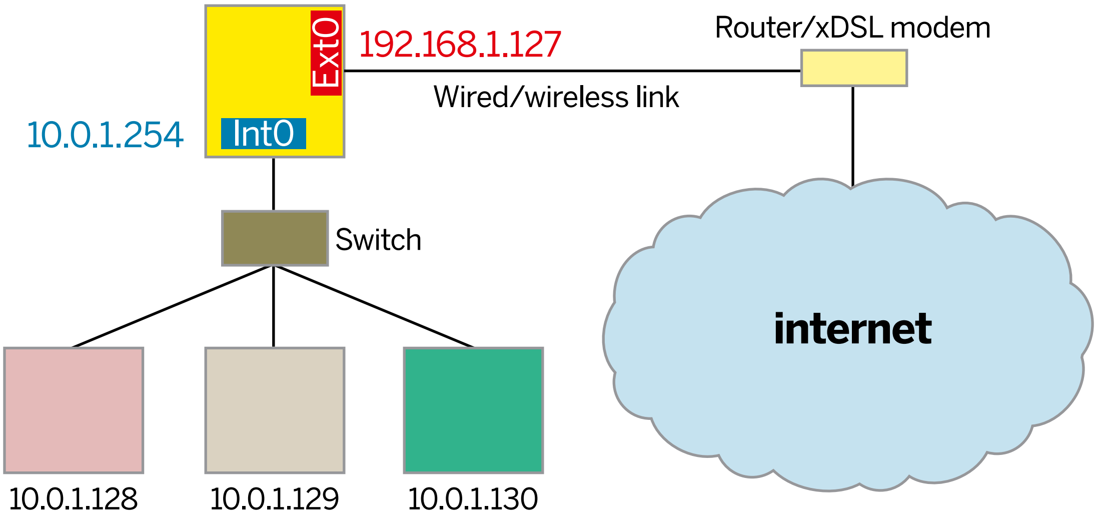
We'll cover setting up a wireless gateway later, but for now assume we've got some computers that aren't connected to the external network, and one computer (the gateway) that is. We'll assume all of these computers are connected by wired connections to a switch and that the gateway machine is happily talking to the internet (either wired or wirelessly – for now we're just assuming that the internal network isn't connecting wirelessly to the gateway, this requires some extra configuring that we'll cover later).

Setting up IP addresses

The first thing we need to do is get the machines talking to each other, which requires getting their IP addresses set up nicely. We won't deal with IPv6 in this tutorial (but soon my pretties), since most home routers still work with IPv4 and it's simpler to keep our protocols homogenous. Your distro may have progressed to kernel-generated persistent names for your network devices, in which case your wired and wireless devices will, have names such as enp0s327 and wlp999, respectively. Or you may still have the old-style, human-readable names, such as eth0 and wlan0. We'll just refer to

Quick tip

Simple traffic routing doesn't require a powerful CPU, but if you're anticipating a lot of traffic you'll want to make sure you've a Gigabit adaptor in the gateway and a gigabit switch.



Thanks to the miracle of NAT, packets can traverse the murky marshes of our internal network, negotiate all kinds of perils in the 192 plains and soar amongst the cloud birds of the wider internet.

the external interface of the gateway (eg the one connecting to your ISP-supplied router) as ext0, and the internal interface (the one connecting to the switch) as int0. So you'll have to make your own substitutions.

RFC1918 provides a few standard IPv4 addresses that can only be used for internal networks, eg 10.x.x.x, 172.16.x.x, 192.168.x.x, so let's go with the first format. We can be a bit more specific here, so let's stipulate that all our internal IPs are in the form of 10.0.1.x. This is usually written 10.0.1.0/24 ie a 24-bit netmask (255.255.255.0). We'll set up the gateway machine's Ethernet interface manually, with the static IP address **10.0.1.254**. Depending on how your machine is set up, this might be achieved through *Network Manager*, the **ifconfig** or **ip** commands, making a *netctl* (Arch Linux) script, editing **/etc/network/interfaces** (Debian). Whatever you choose, it should be straightforward, and also not interfere with the gateway machine's existing internet connection.

We could do this for all the machines on our network, but it's easier to use the *dnsmasq* program, which provides a simple DHCP server to allocate addresses. This program will also be useful later since it will enable the gateway machine to masquerade as a DNS server (hence the program's name).

Installing *dnsmasq* will just be a matter of

```
$ sudo apt-get install dnsmasq
```

or whatever is your distribution's equivalent command is. This will install a heavily-commented **/etc/dnsmasq.conf** file which we shall tweak to our requirements. It's good practice to add the following lines right after their commented equivalents, so that options are sensibly grouped. We'll first set up our ersatz DNS server to listen to requests from our fledgling internal network. Add the following line to **/etc/dnsmasq.conf**:

```
listen-address=10.0.1.254
```

We'll also set up our DHCP server while we've got this file open. It's going to allocate addresses in the range **10.0.1.128** to **10.0.1.253**, this way addresses with a lower final octet can be reserved for machines on the internal network which need static IPs. It's also possible to have DHCP assign specific addresses to specific machines based on their MAC address, using the **dhcp-host** option:

```
dhcp-range=10.0.1.128, 10.0.1.253, 12h
```

Now start the *dnsmasq* service with

```
$ sudo systemctl start dnsmasq
```

or if you're not running systemd:

»

Quick tip

Basic forwarding, routing and masquerading can now be carried out via systemd's networking daemon, networkd. Is nothing sacred? We're taking bets on what its next trick will be. Our money's on Sirystemd: "Siry, can I have my init scripts back, please?" No! – ED.

Hotspots with hostapd

On Debian (and friends) **sudo apt-get install hostapd** will install, besides the *hostapd* program, a gzipped sample configuration file which you can peruse with:

```
$ zcat /usr/share/doc/hostapd/examples/hostapd.conf.gz | less
```

Other distros will install a similar file, probably in another place. This file is heavily commented and gives examples for setting up all manner of access points. We'll only need a few options to get our hotspot up and running, so rather than edit the example file, we'll start afresh. *Hostapd* needs to know where to find this file so **edit /etc/default/hostapd** and add the line:

```
DAEMON_CONF="/etc/hostapd/hostapd.conf"
```

This is a sane place to put the file and may already be the default on your distribution.

Before we proceed any further, make sure that you know the name of your wireless interface (you can check this with **ifconfig**). For this section we'll refer to the wireless network as **wlan0** and the wired external network as **eth0**. A basic WPA2 password protected network is set up with the following **/etc/hostapd/hostapd.conf** file:

```
ssid=LXFwireless
wpa_passphrase=secret passphrase
interface=wlan0
auth_algs=3
channel=6
driver=nl80211
hw_mode=g
logger_stdout=-1
logger_stdout_level=2
max_num_sta=5
rsn_pairwise=CCMP
wpa=2
wpa_key_mgmt=WPA-PSK
wpa_pairwise=TKIP CCMP
```

The **driver** option might require some trial and error – most hardware will work with the **nl80211** driver, but you might need something else here. Test your *hostapd* configuration using the following:

```
$ sudo hostapd -d /etc/hostapd/hostapd.conf
```

If you don't see any errors, then try and connect to your hotspot with your phone. It probably won't get anywhere since there's no DHCP service at this stage, but you should at least be able to authenticate. We'll get DHCP for free once we set up our network bridge.

```
$ sudo service dnsmasq start
```

Now set all the other internal network machines to use DHCP on the local network and you should discover two things: That the internal network machines can all ping each other (by IP address, once you've figured out who's who) and, further, that while they can't ping the outside world, they can at least perform name resolution. *Dnsmasq* will cache queries too, which saves a few milliseconds, should you query the same address more than once. When the client machines obtain a DHCP lease, *dnsmasq* will push the required gateway and DNS settings automatically, and the clients' **/etc/resolv.conf** files will be updated accordingly.

Packet forwarding

In order to get traffic flowing we need to set up packet forwarding and NAT (Network Address Translation). The first thing to do is to enable packet forwarding in the kernel. You can do this through the **/proc** interface like so:

```
$ echo 1 > /proc/sys/net/ipv4/ip_forward
```

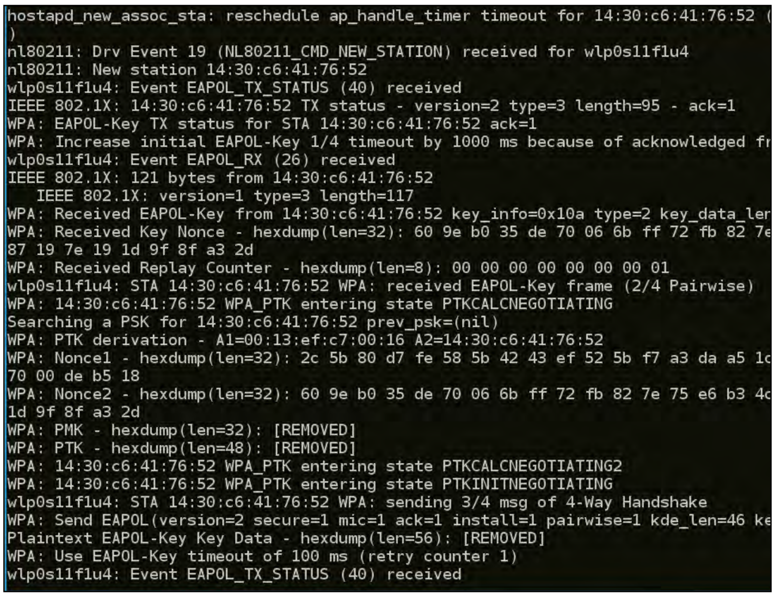
To make this persist across reboots depends on your distro – some still use the **/etc/sysctl.conf** file, so add:

```
net.ipv4.ip_forward=1
```

Some (Arch and derivatives) have deprecated this file in favour of individual fils in **/etc/sysctl.d/**. If this is you create a file, say **/etc/sysctl.d/40-ip-forward.conf**, and add the above line to it. That tells the kernel that forwarding is allowed, but we still have to stipulate the whences and wheretos. We'll use some good old fashioned *iptables* rules to do this. It's certainly possible with the newer *nftables* framework [*see Administeria, p54,* **LXF185**], but with every distro running a different version, and it still being quite new code, we won't do that. Since we're assuming our gateway machine is behind another gateway (that connects to your ISP), we needn't worry too much about security. So we can tell *iptables* to forward packets from the outside to our internal network:

```
$ sudo iptables -A FORWARD -i int0 -o ext0 -j ACCEPT
```

If you haven't previously added any *iptables* rules of your own then its very likely you won't need the previous line – most distros will, by default, allow all packets to be forwarded, as well as allowing everything else. We will need to set up NAT, so that traffic coming out of ext0 can find its way back to int0. This step changes the packets' source addresses to our gateway's IP, and keeps track of the connection so that, as if by magic, any responses are forwarded to the correct machine on our internal network.



```
hostapd_new_assoc_sta: reschedule ap_handle_timer timeout for 14:30:c6:41:76:52 (
)
nl80211: Drv Event 19 (NL80211_CMD_NEW_STATION) received for wlp0s11f1u4
nl80211: New station 14:30:c6:41:76:52
wlp0s11f1u4: Event EAPOL_TX_STATUS (40) received
IEEE 802.1X: 14:30:c6:41:76:52 TX status - version=2 type=3 length=95 - ack=1
WPA: EAPOL-Key TX status for STA 14:30:c6:41:76:52 ack=1
WPA: Increase initial EAPOL-Key 1/4 timeout by 1000 ms because of acknowledged fr
wlp0s11f1u4: Event EAPOL_RX (26) received
IEEE 802.1X: 121 bytes from 14:30:c6:41:76:52
   IEEE 802.1X: version=1 type=3 length=117
WPA: Received EAPOL-Key from 14:30:c6:41:76:52 key_info=0x10a type=2 key_data_ler
WPA: Received Key Nonce - hexdump(len=32): 60 9e b0 35 de 70 06 6b ff 72 fb 82 7e
87 19 7e 19 1d 9f 8f a3 2d
WPA: Received Replay Counter - hexdump(len=8): 00 00 00 00 00 00 00 01
wlp0s11f1u4: STA 14:30:c6:41:76:52 WPA: received EAPOL-Key frame (2/4 Pairwise)
WPA: 14:30:c6:41:76:52 WPA_PTK entering state PTKCALCNEGOTIATING
Searching a PSK for 14:30:c6:41:76:52 prev_psk=(nil)
WPA: PTK derivation - A1=00:13:ef:c7:00:16 A2=14:30:c6:41:76:52
WPA: Nonce1 - hexdump(len=32): 2c 5b 80 d7 fe 58 5b 42 43 ef 52 5b f7 a3 da a5 1c
70 00 de b5 18
WPA: Nonce2 - hexdump(len=32): 60 9e b0 35 de 70 06 6b ff 72 fb 82 7e 75 e6 b3 4c
1d 9f 8f a3 2d
WPA: PMK - hexdump(len=32): [REMOVED]
WPA: PTK - hexdump(len=48): [REMOVED]
WPA: 14:30:c6:41:76:52 WPA_PTK entering state PTKCALCNEGOTIATING2
WPA: 14:30:c6:41:76:52 WPA_PTK entering state PTKINITNEGOTIATING
wlp0s11f1u4: STA 14:30:c6:41:76:52 WPA: sending 3/4 msg of 4-Way Handshake
WPA: Send EAPOL(version=2 secure=1 mic=1 ack=1 install=1 pairwise=1 kde_len=46 ke
Plaintext EAPOL-Key Key Data - hexdump(len=56): [REMOVED]
WPA: Use EAPOL-Key timeout of 100 ms (retry counter 1)
wlp0s11f1u4: Event EAPOL_TX_STATUS (40) received
```

› This is what a successful WPA2 handshake should look like, but our hotspot fails to make it warm in here [*see Hotspots With Hostapd, p81*].

```
$ sudo iptables -t nat -A POSTROUTING -o ext0 -j
MASQUERADE
```

A more robust approach here is to use SNAT (Source NAT) instead of MASQUERADE. This is conditional on the ext0 interface on your gateway having a static IP, though. Note that this has nothing to do with how your ISP assigns IP addresses, merely how you set up your gateway machine's ext0 interface. Using the following line instead means that NAT-ed connections can better survive link loss:

```
$ sudo iptables -t nat -A POSTROUTING -o ext0 -j SNAT
--to-source 192.168.1.127
```

If ext0 does not have the IP address above, then this won't work, funnily enough. The reason this doesn't work with a variable address is that if the link is interrupted then it's possible that upon reconnecting ext0 is assigned a different address. The MASQUERADE target just forgets everything in the event of link loss, so connections can be safely (but slowly) re-established via the new address.

If you're running services on your internal network, and you want these to be accessible from the external network, then you must set up port forwarding, eg if you have an SSH server running on the machine **10.0.1.1**, in order for externally networked machines to see it, they'll need to connect via the gateway machine **192.168.1.254**. Since you might have an SSH server running on the gateway already, we'll forward its TCP port 2222 to port 22 on the internal machine:

```
$ sudo iptables -t nat -A PREROUTING -i ext0 -p tcp --dport
2222 -j DNAT --to 10.0.1.1:22
```

Now you can connect from the external network by pointing your SSH client at your gateway's port 2222. If you wanted to connect from the outside world, then you could add another rule which forwards traffic from your primary router to our gateway machine's port 2222. Once you've got everything working it's good to save your *iptables* rules. Each distro does this slightly differently, but ultimately rules are saved to a text file with:

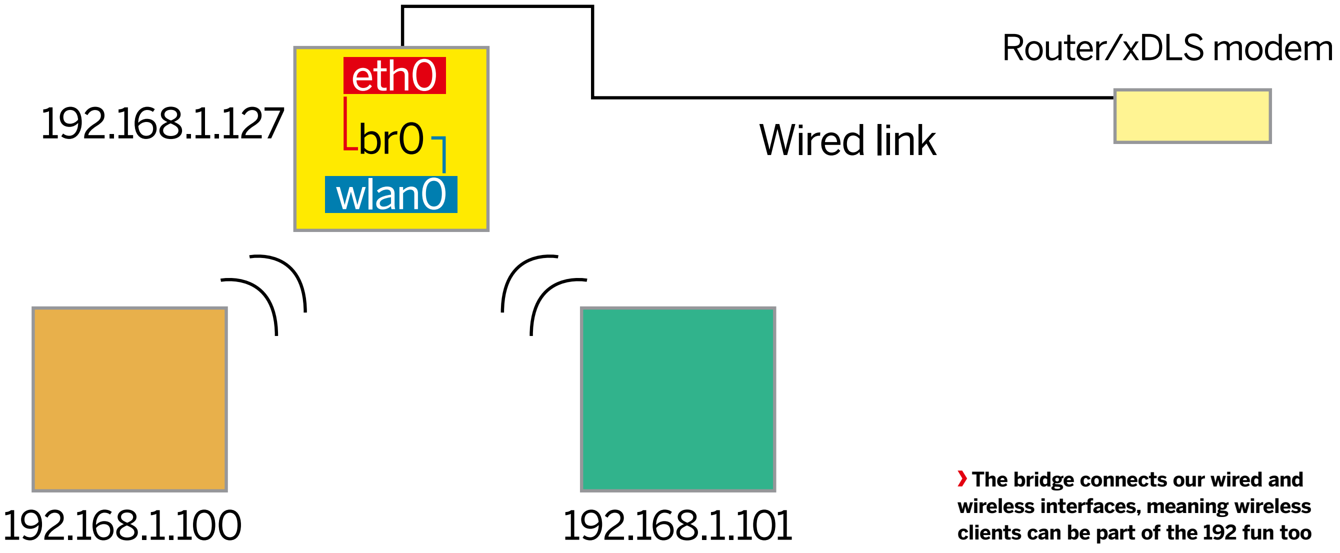
```
$ sudo iptables-save > /etc/iptables.rules
```

Some distributions will auto-magically restore firewall rules from this file, others require to be told. Species of the Debian lineage, for example, would require the following line to be added to **/etc/network/interfaces**:

```
pre-up iptables-restore < /etc/iptables.rules
```

Cutting cords

Sometimes it's desirable to make your router accessible wirelessly, so that your mobile devices can connect to it. This is particularly useful if you find yourself using said devices in areas of your house with poor wireless coverage: Your gateway machine may be connected to your ISP-supplied box via Powerline Ethernet, and situated in or around the blackspot, bringing light to the darkness. It is also possible for this to work if there is a wireless link instead of a Powerline one, so that we have a simple wireless repeater. This may require two wireless cards in the gateway machine though, since some cards cannot be access points and clients



› The bridge connects our wired and wireless interfaces, meaning wireless clients can be part of the 192 fun too

simultaneously. Furthermore, some cards lack Linux support for Access Point mode entirely. To check the capabilities of your wireless adapter use the **iw** tool (**sudo apt-get install iw**, if you don't have it). Running

```
$ iw list
```

will show you everything you could possibly want to know about your wireless devices. Check the 'Supported interface modes' section and ensure that 'AP' features somewhere. For a repeater set up you'll want to check 'valid interface combinations' for 'AP, mesh point'. Bear in mind that mesh protocols haven't yet been standardised so we won't try and cover them here. You'll need to be sure your wireless driver supports 4-address frames if you want to have a go.

Building bridges

So let's reverse our previous setup – we'll have an internet-facing wired connection to the gateway machine and we want to set up a local wireless network which provides access to this. In the previous section we used NAT to direct traffic between the internal and external networks, for this part we'll show you the alternative: network bridging. In many ways this is easier to understand than NAT, since it effectively unifies the two (or more) networks – providing a transparent bridge over which traffic flows unimpeded. However, the NAT approach is a little more flexible and provides more granular control, so you may prefer to use it again.

Whatever your preference, the first task is to install and configure the **hostapd** package (*see the Hotspots With Hostapd box, p81*).

A network bridge amalgamates two disparate interfaces (eg our eth0 and wlan0) into a single interface, which in our example will be br0. Traffic will flow unimpeded between the two interfaces, and machines connected to our wireless hotspot will behave exactly as if they were connected to the same network as eth0. Hopefully, anyway. The first thing we'll need to **sudo apt-get install bridge-utils**, or equivalent. Then we create a new bridge with

```
$ sudo brctl addbr br0
```

and then add our wired interface to it with:

```
$ sudo brctl addif br0 eth0
```

We can't add the wlan0 interface to the bridge without first starting our access point, since only devices in so-called promiscuous mode can be bridged. We can automate this by adding the following line to our **hostapd.conf**:

```
bridge=br0
```

Now we can test our access point by starting the service:

```
$ sudo service hostapd start
```

Replace with **systemctl start hostapt** if you're using Systemd. If everything works then enable the service. You can recreate the bridge setup in *Network Manager*, or by modifying **/etc/network/interfaces**.

And that concludes our foray into the world of bespoke routers. We've covered two different set ups, but don't be afraid to mix and match if you require, all these technologies will work together where it makes sense. We'll also be covering replacing the firmware on your home router with Linux one of these days. For now we're all huddled round an overclocked Raspberry Pi for Wi-Fi warmth. LXF

Next issue: Zentyal server

Anonymising gateway

A nice thing about this setup is that any routing beyond the gateway machine propagates trivially to our internal network. So if you have your gateway machine's traffic routed through a VPN or *Tor*, then so can all our internal network traffic. Installing *Tor* is straightforward, the default config sets up a SOCKS proxy which you must connect your applications to. You can make this proxy available to the internal network by adding a line, such as **SOCKSPort 10.1.0.254:9050** and **TransPort 10.1.0.254:9040** to **/etc/tor/torrc**. Then you can set up any applications running on our internal network to use that proxy. A more passive approach is to set the gateway machine up as a transparent proxy or an isolating proxy, but this is beyond the scope of this little box.

You may subscribe to a commercial VPN provider, or even be running your own OpenVPN server somewhere on the internet. Either way, connection is usually achieved by setting up a TUN device (usually called tun0) on the client end. The routing table is then modified so that all traffic travels via tun0, redirecting everything through the VPN. If you have already successfully set this up on your gateway machine, then all that is required is a small addition to our NAT rules:

```
$ sudo iptables -t nat -A POSTROUTING -o tun0 -j MASQUERADE
```

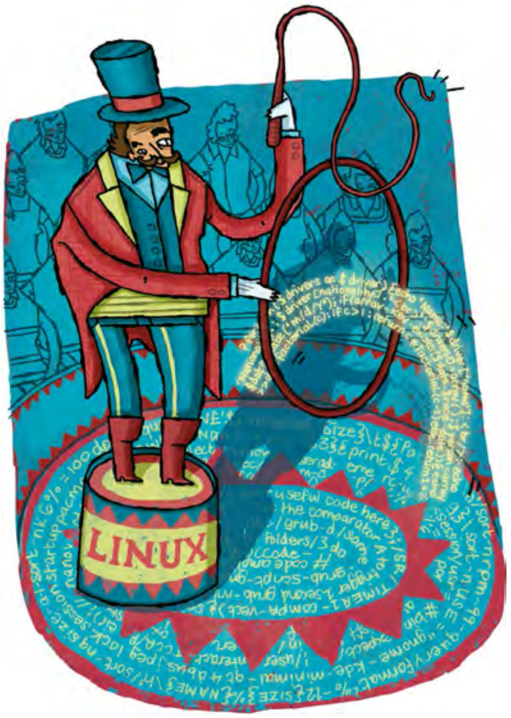
Since VPN traffic travels through the tun0 interface, this line ensures that it can still find its way back to the internal network. As before, you can use SNAT here if ext0 has a static IP.

System coding: the Linux kernel

Dr. Chris Brown kicks off a new series on systems programming, surely the most fun you can have without taking your clothes off.

Our expert

Dr. Chris Brown provides Linux training, authoring and consultancy. He finds his Ph.D. in particle physics to be of no help in this work at all.



We're going to look at the Linux kernel through a programmer's eyes, in this series. We'll be examining the system calls that allow our programs to obtain services from the kernel, and also some of the interesting bits of the standard library. Traditionally, these things are done using C, but C is not the only language to expose the system call interface, and to prove the point, we'll present some of our examples in Python. Along the way, we'll develop a few simple programs ranging from a shell to a web server.

Kernel space and user space

To get started, let's talk architecture for a moment. As you know, the kernel is the real heart of Linux. It provides services such as memory management, process scheduling, the file system, and the TCP/IP network stack. It implements access controls based on process identity and file permissions, and also provides the modules (sometimes called device drivers) that manage the actual hardware. All of this software runs in a privileged processor mode. We call this 'kernel space'. All other programs – the shell, command line tools, graphical apps – everything, run in an unprivileged mode. We say they run in 'user space'. The kernel provides its services to the user space programs through a set of tightly defined entry points known as system calls. At the last count there are around 350 of these and they provide services ranging from accessing files to creating processes and network sockets. It's this system call interface that's the focus of these tutorials.

Now it turns out that programs don't make system calls directly, they do so via thin 'wrapper' routines in the standard library, glibc [*see diagram, p85*]. Taking the **write()** call as an example, there's a little function called **write()** in glibc that simply marshals the arguments in the right way and does the little bit of magic necessary to make the jump into kernel space. As another example, consider the familiar **printf()** routine. This isn't a system call; all the fancy formatting it does occurs in user space. It (presumably) eventually calls **write()** to actually push the resulting byte stream out. Other library routines, such as **sqrt()** for example, operate entirely in user space and return their result to the program without ever diving in to the kernel.

The waters sometimes get a little muddier. Things that used to be system calls in unix, such as **exit()**, have become library routines in Linux, calling down to even lower level system calls that you're not expected to invoke directly. But to be honest, as a programmer you don't much care whether something is a system call or a library routine, except that the system calls are documented in section two of the man pages and library routines are in section three. So

```
$ man 2 write
```

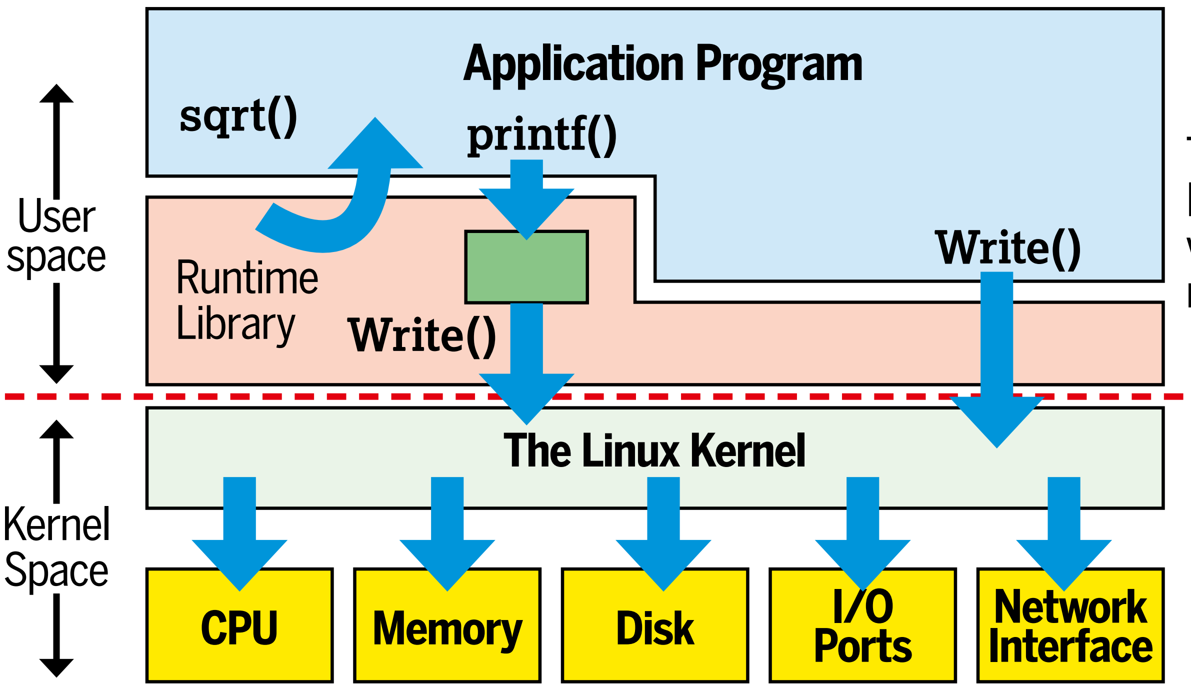
gets you the man page for the write() system call, whereas

```
$ man 3 printf
```

gets you the printf library routine.

To get started, we're going to look at four (yes, four) ways to copy a file. Here's the first one. It demonstrates direct use of system calls and the lowest level approach of the four:

```
/* Trivial file copy program using low-level I/O */

#include <fcntl.h>
#include <stdlib.h>
#define BSIZE 16384

void main()
{
  int fin, fout; /* Input and output handles */
   char buf[BSIZE];
   int count;

   if ((fin  = open("foo", O_RDONLY)) < 0) {
     perror("foo");
```



› System calls provide the entry points from user space to kernel space. In reality, all system calls are made via library wrappers.

```
15.     exit(1);
16.   }
17.   if ((fout = open("bar", O_WRONLY | O_CREAT, 0644)) <
0) {
18.     perror("bar");
19.     exit(2);
20.   }
21.   while ((count = read(fin, buf, BSIZE)) > 0)
22.     write(fout, buf, count);
23.
24.   close(fin);
25.   close(fout);
26. }
```

This program illustrates five system calls: **open()**, **read()**, **write()**, **close()** and **exit()**, and one library routine, **perror()**. There's quite a lot going on here and we'll dissect the code in detail. Lines 3 and 4 include a couple of header files. Typically, these contain function prototypes and definitions of symbolic constants (such as **O_RDONLY** in this example). How do you know what files to include? The man pages will tell you. (*See Reading the Man Pages, below.*) The compiler will throw up errors if you omit a header file, but it won't tell you which you've forgotten. At line 5 we define the size of the buffer we're going to use during the copying process. Choosing a large number here will speed things up by reducing the number of read() and write() calls the program will make, and choosing a multiple of the file system's block size (typically 4k) will also help. The buffer is actually declared at line 10.

The **main()** function at line 7 is the entry point of a C program. At line 13 we meet **open()** – our first system call. The first argument **"foo"** is the input file name; it's a relative path name (it doesn't start with a '/') so Linux will interpret the name relative to the current directory that the program is running in. Alternatively I could have used an absolute path name like **"/home/chris/demo/foo"**. Now, **open()** returns a 'file handle' (a plain old integer) which we assign to fin. Hard-wiring the file names into the code is clearly dumb; later in the series we'll see how to pluck these from the program's command line.

Exceptions and failures

If you're coming from .NET or Java, you're used to having methods raise exceptions when they run into trouble. The Linux system call interface doesn't use exceptions; instead, calls like **open()** which normally return a positive integer will return -1 to indicate an error, for example if the file "foo" »

Quick tip

The best book on this stuff is *The Linux Programming Interface* by Michael Kerrisk. It's head and shoulders above the others, but at 1,500 pages, the word 'tome' comes to mind.

Reading the man pages

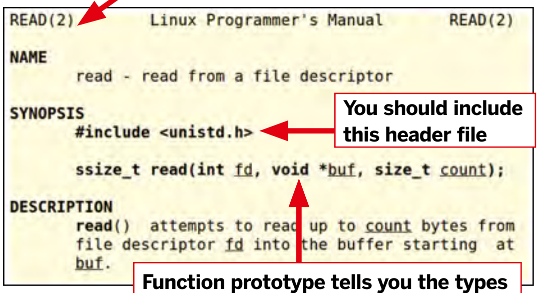
The man pages that describe the system calls can be a little intimidating, and you need to understand how to read them. The annotated screen shot of the man page for **read()** should help, but I want to make one point clear: the line of code you see in the man page is *not* an example of a call to the function; it's the function's prototype. For example, you might be tempted to write:

```
void *p;
read(0, p, 10);
```

Here we're consuming 10 bytes from file descriptor 0 (standard input). Well, the types all match up with what the man page says, and the code compiles OK. The problem is that p isn't actually pointing to an allocated buffer! We need to either allocate the buffer at compile time, or allocate it at run time like this:

```
char *p;
p = malloc(10);
read(0, p, 10);
```

The 2 means it's a system call – 3 means it's a library routine.



```
READ(2)          Linux Programmer's Manual          READ(2)

NAME
       read - read from a file descriptor

SYNOPSIS
       #include <unistd.h>

       ssize_t read(int fd, void *buf, size_t count);

DESCRIPTION
       read() attempts to read up to count bytes from
       file descriptor fd into the buffer starting  at
       buf.
```

You should include this header file

Function prototype tells you the types of the parameters and the return type.

» doesn't exist. You're expected to check the return value from the call and respond appropriately. Line 13 illustrates why I love C so much. In C, an assignment such as a = b not only has a 'side effect' (changing the value of a) it also has a value (the value of b). It's this value that's being tested in the **if** statement on line 13. So this one line of code is really doing three things: opening the file, saving the resulting file descriptor, and testing this value to check if the call failed.

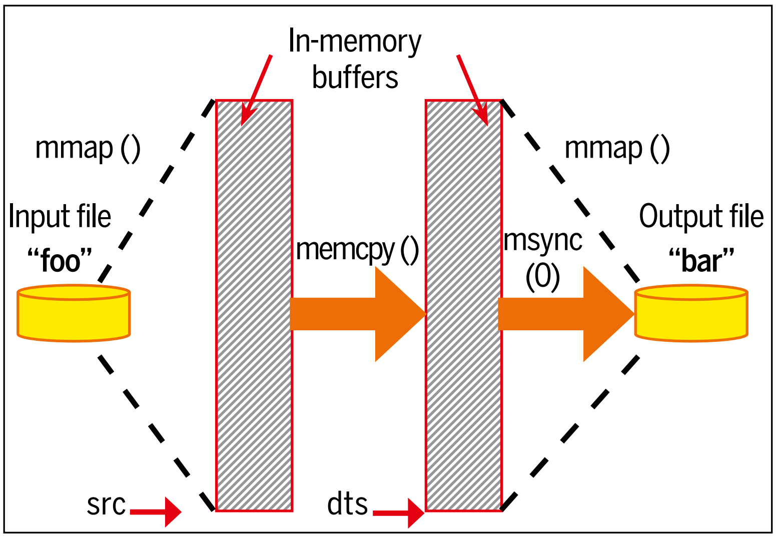
When a system call fails, it sets the value of a global integer variable called errno to indicate the cause of the error. For example a value of 2 means 'no such file or process' and 13 means 'permission denied'. However, best practice is to use symbolic constants like ENOENT and EACCESS instead of numeric values. We don't reference errno explicitly in our code, but it's used by the library routine **perror()** at line 14 to index into a table of error messages. The appropriate message is printed on the standard error stream. After that, at line 15 we exit the program, returning a non-zero exit code to indicate that we ran into trouble. I'll talk about exit codes later in the series when I discuss processes. Lines 17-20 do much the same thing to open the output file.

Notice that this is not an object-oriented model. The **open()** call does not return some sort of file object on which we can invoke read and write methods; instead it returns a plain integer file handle, which can be passed as a parameter in subsequent read() and write() calls. Lines 21 and 22 are the heart of this program. Line 21 is another example of the 'do something, capture the result, and test it' idiom that we met earlier. Here, the 'do something' part is to read up to BSIZE bytes into the buffer buf. It's important that the buffer you're reading into is at least as large as the number of bytes you're asking to read. The return value from the **read()** call is the number of bytes we actually got. Unless we've reached the end of the file, this will be the same as the number we asked for. For example, if the file had exactly 40,000 bytes we would do four reads, which would return 16384, 16384, 7232 and 0 respectively. The fourth read, of course, would cause the loop to terminate. At line 22, we write however many bytes we got to the output file. Notice that C has no boolean data type – it uses integers instead, with nonzero meaning true and zero meaning false. So you can write line 21 more compactly:

```
while (count = read(fin, buf, BSIZE))
```

It's common practice for C programmers to hide all the important actions as side effects in evaluating the test predicate for an if or while statement, as we do here.

› **The mmap() system call lets you read and write a file as if it were an in-memory array.**



Once the copy loop has finished, we close our two streams at lines 24 and 25. In this particular example it doesn't really matter because the program terminates immediately afterwards and its open streams will be implicitly closed. However, it's good practice to close desciptors when you're done with them, because there's a limit on how many the process can have open and a long running program (say a server that opens a file every time a client connects) will eventually run out if it doesn't close them.

Maybe you're thinking that this system programming stuff is very low-level. You're right. Short of crawling out over the surface of the disk with a tiny bar magnet, you can't do I/O at a lower level than this in Linux.

Remember Tim Toady?

Of course, there's always more than one way to do it. Traditionally, the system call interface is discussed using the C language. The man pages all show C function prototypes, and C remains the lingua franca of Linux systems programming. But there are other languages that provide language bindings to the self-same set of system calls. Take python for example – the "os" module provides lots of functions that provide direct access to the system calls and parallel the "C" calls almost one for one. To prove the point, here's our file copy program in Python:

```
import os

bsize = 16384
fin  = os.open("foo", os.O_RDONLY)
fout = os.open("bar", os.O_WRONLY | os.O_CREAT, 0o644)

while 1:
  buf = os.read(fin, bsize)
  if buf:
    os.write(fout, buf)
  else:
    break

os.close(fin)
os.close(fout)
```

I'm sure that the pythonistas among you are queuing up to point out that there are 'better' ways to do this, and you would be right. I've deliberately chosen an approach that parallels the C version as closely as possible. I won't do another line-by-line breakdown, but for any readers fluent in C but new to Python, here are some key differences:

» **A dynamically typed language** We don't have to pre-declare variables such as fin and buf. They spring into existence (and take on a type) at the point at which you assign to them.

» **No curly brackets** Python doesn't use them to delimit loops and branches, it uses indentation.

» **No pre-allocated buffer** When passed to the read() call. Instead, the call returns a bytestring, a built-in Python type analogous to an array of char in C, but which knows how long it is. So the code does not directly expose anything equivalent to the count variable that we had before.

» **Run-time error reporting** I haven't included code to detect and report errors in the os.open() calls, because the default behaviour of Python when it encounters run-time errors is mostly equivalent to the perror() calls I included in the C code.

» **Different syntax for octal constants** The strange notation 0o644 is not a typo. As from Python 3.0, it's the syntax for octal constants. Yes, really.

Typedef name	Actual Type	Description
pid_t	int	A process ID or process group ID
gid_t	unsigned int	A numeric group identifier
uid_t	unsigned int	A numeric user identifier
time_t	long int	Time (in seconds) since "the epoch"
size_t	unsigned long	The size of an object in bytes
ssize_t	long int	The size of an object, or a negative error indication
mode_t	unsigned int	File permissions
off_t	long int	A file offset or size
socklen_t	unsigned int	The size of a socket address structure

› **For code portability, most data types used by system calls are defined using typedefs. Here are a few.**

Let's get portable

Our third implementation of the file copier rises above the system call level to use the standard library:

```
#include <stdio.h>
#define BSIZE 16384

void main()
{
   FILE *fin, *fout;
   char buf[BSIZE];
   int count;

   fin = fopen("foo", "r");
   fout = fopen("bar", "w");

   while ((count = fread(buf, 1, BSIZE, fin)) > 0)
      fwrite(buf, 1, count, fout);

   fclose(fin);
   fclose(fout);
}
```

It doesn't look much different really, does it? Notice that open(), read(), write() and close() are replaced by **fopen()**, **fread()**, **fwrite()** and **fclose()**, and file descriptors are of type **"FILE *"** rather than plain integers. The crucial difference is that our previous example used system calls that are specific to Linux and Unix; here, we're using routines that are part of standard C. Any implementation of C that conforms to the ANSI standard should provide them, regardless of the underlying operating system. So this version is likely to be much more portable.

Something completely different

Our final copy program is very different. It uses the **mmap()** system call to map the input and output files into memory, then simply does a memory-to-memory copy.

```
 1. #include <sys/mman.h>
 2. #include <sys/stat.h>
 3. #include <fcntl.h>
 4. #include <stdlib.h>
 5. #include <string.h>
 6.
 7. int main()
 8. {
 9.   char *src, *dst;
10.   int fin, fout;
11.   struct stat sb;
12.
13.   fin = open("foo", O_RDONLY);
14.   fstat(fin, &sb);
15.   src = mmap(NULL, sb.st_size, PROT_READ, MAP_
PRIVATE, fin, 0);
16.
17.   fout = open("bar", O_RDWR | O_CREAT | O_TRUNC,
S_IRUSR | S_IWUSR);
18.   ftruncate(fout, sb.st_size);
19.   dst = mmap(NULL, sb.st_size, PROT_READ | PROT_
WRITE, MAP_SHARED, fout, 0);
20.
21.   memcpy(dst, src, sb.st_size);
22.   msync(dst, sb.st_size, MS_SYNC);
23.
24.   exit(0);
25. }
```

Bits of this should look familiar by now, but some of it is very new. The **fstat()** system call at line 14 reads the input file's attributes into a 'stat' structure (sb); the only field we're interested in here is **st_size**, the file's size. Line 15 is really interesting. It maps the file into memory. We capture the return value from **mmap()**, which is the address at which it mapped the file, rather like the way **malloc()** allocates a lump of memory and returns a pointer to tell you where it is. Once this is done, we can access the data in the file as if it were any array: **src[0]** gets us the first byte of the file, and so on. I will leave you to look up the man page on mmap to figure out exactly what all the other arguments do. Similarly, lines 17-19 map in the output file, setting its size (via **ftruncate()**) to match the input file. All the real work is done by line 21 -- this is a pure in-memory copy of the two mapped files. Finally at line 22, **msync()** flushes the changes in the **dst** array back out to the file. Magic! Next month, we'll tackle random file access using **lseek()** and friends, and see how to manage links and directories. See you then! LXF

Data types

The Linux system call interface specifies a large number of data types, defined using typedefs in the various header files. Depending on your point of view (and degree of cynicism) the purpose of these is either to make the code more portable, or to make it harder to understand. For example, you'll discover from the man page that the **read()** call returns an **ssize_t**. What the heck is that? Chasing through the chain of include files to find out can be tricky. In the case of **ssize_t** we can (if we dig deep enough) uncover two **typedefs**:

```
typedef long int __ssize_t;
typedef __ssize_t ssize_t;
```

So an **ssize_t** is really just a long -- at least on my machine. In fact, most of these data types turn out simply to be either signed or unsigned integers. I have included a table of the most common, though it is by no means a complete list. The entries in the 'actual data type' column reflect my own 64-bit Linux installation; your mileage may vary. The point is that you are not supposed to care what the actual underlying type is; you just declare variable of types that the man pages tell you to use, and bask in the glory of knowing your code will be more portable as a result.

MariaDB: get to grips with SQL

Jonni Bidwell turns the tables on Oracle's MySQL as he dabbles in the wholesome fork that is MariaDB and claps his hands like a seal.

Our expert

Jonni Bidwell feels that Spring is in the air, time to get out and enjoy the outdoors. Right after he updates his Gentoo box.

The *MySQL* RDBMS (Relational database management system) first appeared on the scene in 1995, and its then custodian, Swedish firm MySQL AB, became one of the earliest examples of a successful open source company. While the database was entirely available under the GPL, clients unwilling to pander to its clauses (ie those who wanted to implement *MySQL* databases in closed-source products) could acquire the software under a monetary licence. In 2008, MySQL AB was acquired by Sun Microsystems, giving them a competitor to arch rival Oracle's database. However, two years hence, Sun was acquired by Oracle, so that *MySQL* was now, in a sense, in the hands of the enemy. As one may imagine consternation ensued in the *MySQL* community, the main concern being that, under Oracle's stewardship, *MySQL* would become a closed (or at least partially so) product. The very day that the takeover was announced, Michael Widenius (former CTO of MySQL AB) and a number of other *MySQL* developers forked *MySQL*, and the result was *MariaDB*.

Since taking the helm Oracle has certainly added some closed source extensions to *MySQL*, but it also continues to develop the Community Edition alongside the Enterprise offering. Many in the industry, Canonical in particular, are happy with the attention Oracle continues to give to *MySQL*. In a *ZD-Net* interview, Mark Shuttleworth (Founder of Canonical) dismissed the surrounding exodus as a symptom of the "tendency to imagine conspiracies" on the part of certain players in the open source community.

While *MariaDB* is nowhere near as popular as its rival Oracle (the world's most used RDBMS), its parent *MySQL*, or Microsoft's SQL Server (numbers 2 and 3 on the podium), it is certainly gaining ground. A ranking on **db-engines.com** puts it at #26 as of Feb 2015. Being (almost) a drop-in replacement for *MySQL*, but at the same time offering improved performance (and a more wholesome provenance), means that obstacles to adoption are low. Many distros have already adopted *MariaDB* as their default *MySQL* implementation, among the first to do so were Arch and OpenSUSE 12.3 (March 2013), followed by Fedora 19 (July 2013), now joined by Mageia, Chakra and RHEL 7. Most other distros, while still having packages for Oracle's *MySQL* Community Server, will also have *MariaDB* packages that you can use instead.

This Strange Engine

The guts of any RDBMS, as in the gubbins that does all the CRUD (creating, reading, updating and deleting of data) is known as the database engine. Older versions of *MySQL* used the MyISAM engine, but this was later replaced with InnoDB. Up until version 10.0.9 *MariaDB* also used InnoDB, but it has since then replaced it with a fork called XtraDB, which is a drop-in replacement for InnoDB with some extra features. XtraDB is maintained by Percona, who maintain Percona Server, which is, confusingly enough, another drop-in replacement for *MySQL*.

XtraDB takes greater advantage of modern hardware, offers better scalability, is highly configurable and more memory-efficient. What's not to like? Of course, there are many other database engines supported by *MariaDB* and many places where it's technically superior to its parent. You can read about them on the *MariaDB* website (see **http://bit.ly/MariadbVSMySQL)**.

The first stable version of *MariaDB* to be released, in February 2010, was numbered 5.1.42, in an apparent defiance of convention. The rationale here is that it was based on the then current 5.1.42 version of *MySQL*. Version 5.5, was likewise based on *MySQL 5.5*. However, since the current stable branch of *MariaDB* evolved independently of *MySQL*

Quick tip

For experimenting with compatibility between different versions of MySQL/MariaDB, checkout the handy MySQL Sandbox: **http://mysqlsandbox.net**

5.6, it's numbered differently – they chose a nice round 10.0 for this series. To be clear, this is not to say that *MariaDB 10.0* isn't compatible with *MySQL 5.6* as that would be almost entirely untrue. Rather, many of the extra features and bugfixes introduced by Oracle in 5.6 had already been implemented by the community, and the extensive refactoring of code undertaken by Oracle was deemed unnecessary to replicate. *MariaDB* is its own entity, and has added many of its own features and revamped many under-the-hood components, some of which we'll explore here. However, *MySQL* is also its own thing and while new features added to it will be reflected in *MariaDB* where appropriate, this will take time and so there are edge-cases where the two are incompatible, and this gap may widen in the future.

That said, this is an introductory tutorial and so most of it will work just fine in both the *Maria* and *My* databases (these are the names of Widenius' daughters incidentally, he also has a son called Max, and another database called *MaxDB*). Actually it'll mostly work for any SQL database.

The box [*see p90*] gives some hopefully distro-agnostic instructions for installing *MariaDB*, but do check your own distro's documentation in case some new-fangled install method has been concocted since this was written. Once you've got it installed, the next thing is to set up the database server, a helpful script is provided for this purpose. It needs to be run as root:

```
# mysql_secure_installation
```

This script (which might be run automatically on install, in which case there's no need to run it again) will prompt for the current *MySQL* root password (not your system's root password) which will be blank if you've just done a clean install. You will then be prompted to set a new root password; offered the opportunity to delete anonymous users and test databases, and restrict root access to local logins. All of which you should do, unless you have some reason not to.

We can start the *MariaDB* shell as the root user with:

```
$ mysql -u root -p
```

The **-p** means that we will be prompted for the password. Once we're in, you should see the *MariaDB* prompt **MariaDB [(none)]>**. SQL (Structured Query Language) is a fairly easy to follow language, don't forget the semicolons at the end of every command though. We can see any databases currently in service with a simple:

```
SHOW DATABASES;
```

While the uppercase is optional, and hard on the eyes, you should still use it for SQL commands as it helps to differentiate reserved words from database names and the like. You'll see three databases here, one called **mysql** which is to do with *MariaDB*'s internals and not for you to touch, as well as **information_schema** and **performance_schema** which hold metadata about databases and can be used to diagnose issues. We can create a new database called **lxfdata** and then connect to it with:

```
CREATE DATABASE lxfdata;
USE lxfdata;
```

After the first command, you'll see a friendly acknowledgement that everything went OK:

```
Query OK, 1 row affected (0.00 sec)
```

And note that the prompt changes to **MariaDB [(lxfdata)]>** after the second. You can also tack the database name onto the end of the **mysql** shell command to dive straight into it.

Inside the database, data is stored in tables, but we don't have any yet. So let's rectify this by creating an example table called linuxes, which will house some data about various Linux distros. Tables are formed of columns, and each column has an associated data type. SQL queries will span newlines so you can format them nicely.

```
CREATE TABLE linuxes (
    id int(5) NOT NULL AUTO_INCREMENT,
    name varchar(32) DEFAULT NULL,
    current_version varchar(32) DEFAULT NULL,
    easy bool DEFAULT NULL,
    PRIMARY KEY(id)
);
```

We've set up four columns, here's how we'll use them:

» **id** An id number, this will automatically incremented for each entry.

» **name** The name of the distro, a 32 character string, defaults to the value NULL if not stated.

» **current_version** The distro's version number, or codename. Same datatype as **name**.

» **easy** Whether or not the distro is suitable for beginners. A boolean (true or false) value.

We'll use **id** as a primary key, which is a canonical index for the table. Each table must have at least one primary key and said keys must be unique. You can quit the *MariaDB* shall with **quit** or Ctrl-D.

The screenshot [*pictured below*] shows a summary of our empty database. Booleans are referred to as **tinyints**, since they are stored as the integers 1 or 0. You can read all about the manifold datatypes in *MariaDB* at **http://bit.ly/MariaDBDataTypes** . For example, the **blob** datatype stores up to (but not including) 64K of binary data, and has its counterparts **tinyblob**, for storing up to 256 bytes, as well as **mediumblob** (16MB) and **longblob** (4GB).

Don't hurt yourself

Just like the situation for the rest of Linux, it's generally not a good idea to use the **root** user except when necessary. This is especially true if your database is connected to a web application – if this connects as the root user then all your databases could be at risk if that application becomes compromised. So lets create a less privileged user. First reconnect to our **lxfdata** database:

```
$ mysql -u root lxfdata -p
```

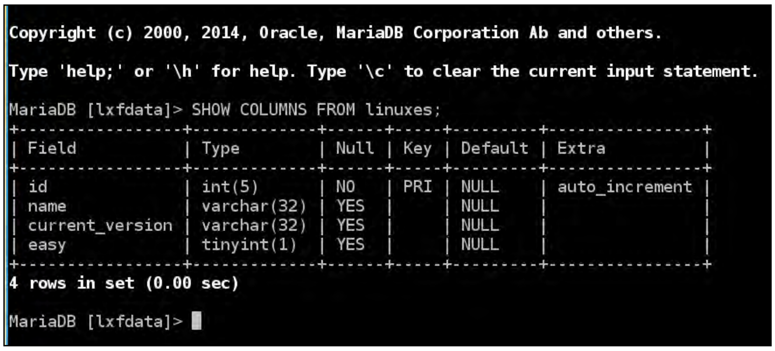
We'll call our user **lxfuser** and give them a weak password:

```
CREATE USER 'lxfuser'@'localhost' IDENTIFIED BY
'password1';
```

»

Quick tip

One of the big names that have switched from *MySQL* to *MariaDB* is Wikipedia. You can read all about its transition here **http://bit.ly/WikipediaAdopts MariaDB**.



```
Copyright (c) 2000, 2014, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [lxfdata]> SHOW COLUMNS FROM linuxes;
+-----------------+-------------+------+-----+---------+----------------+
| Field           | Type        | Null | Key | Default | Extra          |
+-----------------+-------------+------+-----+---------+----------------+
| id              | int(5)      | NO   | PRI | NULL    | auto_increment |
| name            | varchar(32) | YES  |     | NULL    |                |
| current_version | varchar(32) | YES  |     | NULL    |                |
| easy            | tinyint(1)  | YES  |     | NULL    |                |
+-----------------+-------------+------+-----+---------+----------------+
4 rows in set (0.00 sec)

MariaDB [lxfdata]> █
```

› The columns of our linuxes table, you can extend this to include all kinds of data – even, using the blob types, binary data, eg A logo image for each distro.

» The **@ 'localhost'** syntax means this user can only log in locally. You can also specify any hostname here, or use **@'%'** to enable the user to login from anywhere else. Let's say, for instance, we want to allow **lxfuser** to add rows to our table, and select (view) it:

```
GRANT INSERT, SELECT on lxfdata.linuxes TO
'lxfuser'@'localhost';
```

Now we can connect to our database as **lxfuser**, and start adding some data. So exit the shell and log in again with:

```
$ mysql -u lxfuser lxfdata -p
```

Each entry in our table (ie each distro) will occupy a row. Our first row will be the distro of choice at **LXF** towers, the DPRK's Red Star Linux (the OSX-ey Desktop version is now floating around the darker corners of the internets **http://bit.ly/RedStar3Desktop**:

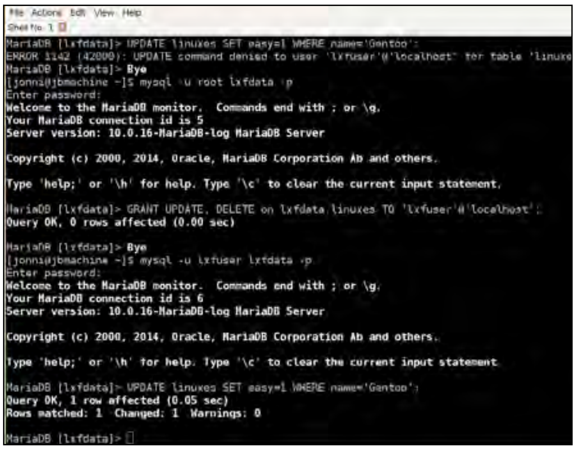
```
INSERT INTO linuxes (name, current_version, easy)
VALUES ('Red Star Linux', '3.0', 0);
```

Red Star is definitely not easy if you don't read Korean. Note that we don't need to specify the **id** column since it auto-increments. Other columns which we don't specify in the first set of brackets will get the value **NULL**, or whatever other default value you specified above. Thus, for our next two favourite distros, which are rolling releases and hence have no version number, we can do:

```
INSERT INTO linuxes (name, easy) VALUES
   ('Arch Linux', 0),
   ('Gentoo', 0);
```

And we conclude this exercise in data entry with some more Linuxes we love:

```
INSERT INTO linuxes (name, current_version, easy)
VALUES
   ('Ubuntu', '14.10 Utopic Unicorn', 1),
   ('Linux Mint', '17.1 Rebecca', 1),
   ('Debian', '7.8 Wheezy', 0),
   ('openSUSE', '13.2', 1),
   ('Fedora', '21', 0),
   ('Mageia', '4.1', 1),
   ('Elementary OS', '0.2 Luna', 1),
   ('Bodhi Linux', '3.0.0', 1);
```



```
MariaDB [lxfdata]> UPDATE linuxes SET easy=1 WHERE name='Gentoo';
ERROR 1142 (42000): UPDATE command denied to user 'lxfuser'@'localhost' for table 'linux
MariaDB [lxfdata]> Bye
[jonni@jbmachine ~]$ mysql -u root lxfdata -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 5
Server version: 10.0.16-MariaDB-log MariaDB Server

Copyright (c) 2000, 2014, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [lxfdata]> GRANT UPDATE, DELETE on lxfdata.linuxes TO 'lxfuser'@'localhost';
Query OK, 0 rows affected (0.00 sec)

MariaDB [lxfdata]> Bye
[jonni@jbmachine ~]$ mysql -u lxfuser lxfdata -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 6
Server version: 10.0.16-MariaDB-log MariaDB Server

Copyright (c) 2000, 2014, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement

MariaDB [lxfdata]> UPDATE linuxes SET easy=1 WHERE name='Gentoo';
Query OK, 1 row affected (0.05 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MariaDB [lxfdata]> 
```

› **We empowered our user and all they did was mess with our data. Tsk.**

We can view the fruits of our labour using the following SQL query:

```
SELECT * FROM linuxes;
```

As you can imagine, tables can become unwieldy and viewing them in their entirety like this will not always be possible, or pleasant. It's handy, then to restrict the rows and columns that are displayed. Suppose we are only interested in the names and version numbers of our distros. This query will show only those columns:

```
SELECT name, current_version FROM linuxes;
```

On the other hand, we might only be interested in 'easy' distributions, which we can display with:

```
SELECT * FROM linuxes WHERE easy = 1;
```

We can also specify a sort order here, rather than the order in which the data were entered. All we need do is add an **ORDER BY** directive, for example to sort the last query lexicographically by distribution name, do:

```
SELECT * FROM linuxes WHERE easy = 1 ORDER BY
name;
```

Queries can be nested, and can include regular expressions via the **REGEXP** operator.

We never granted our humble **lxfuser** the privileges required to change or delete rows, but we can do that if we first logout of the client shell and, while logged in as the root user we do:

```
GRANT UPDATE, DELETE on lxfdata.linuxes TO
'lxfuser'@'localhost';
```

Now if we log in to the client as **lxfuser** we can start messing with our data. We might, if we were to follow the advice of 4chan's **/g/** board, get the idea that Gentoo is easy. We can update the database to reflect our new opinion with:

```
UPDATE linuxes SET easy=1 WHERE name='Gentoo';
```

Deleting rows is straightforward, we could delete Ubuntu (not that we have anything against it, after all Canonical pay our wages, if you believe the whispers) with a simple:

```
DELETE FROM linuxes WHERE name = 'Ubuntu';
```

Columns can be deleted (technically **DROP**-ped) too, but this requires a different privilege, **ALTER**. The second most destructive thing one can do with your database is

Installation differences

As mentioned elsewhere, a few of the distros including Fedora, Arch, openSUSE and some others have chosen *MariaDB* as their default RDBMS, meaning that there will be a **mariadb** package (and on Fedora a **mariadb-server** package as well) waiting in the repositories for you. Ubuntu has chosen to stick with *MySQL*, so the following code will add the MariaDB repo and install everything:

```
$ sudo apt-get install software-properties-common
$ sudo apt-key adv --recv-keys --keyserver hkp://keyserver.ubuntu.com:80 $ 0xcbcb082a1bb943db
$ sudo add-apt-repository 'deb http://mirrors.coreix.net/mariadb/repo/10.0/ubuntu utopic main'
$ sudo apt-get update
$ sudo apt-get install mariadb-server
```

This will also work for Linux Mint. For Debian the procedure is much the same, just replace **software-properties-common** with **python-software-properties** in the first command and change the repository in the third to something like **'deb http://mirrors.coreix.net/mariadb/repo/10.0/debian wheezy main'**. For other distributions and bleeding-edge releases from *MariaDB*'s own repositories, visit **https://downloads.mariadb.org/mariadb/repositories**.

Once installed the service will automatically start on some distros, and other distros, however, you will need to start it manually. For *Systemd* peeps, this will be done with the command **systemctl start mysqld** – *MariaDB* is so compatible that it didn't even change the name of the daemon here.

DROP one or more tables, this requires DROP privileges:

```
DROP TABLE linuxes;
```

The most destructive thing that we can do is to callously delete whole databases:

```
DROP DATABASE lxfdata;
```

In some cases it's possible to (partially) recover accidentally dropped tables, but it's not something you can count on. Rather, you should keep regular backups, which we shall now explore.

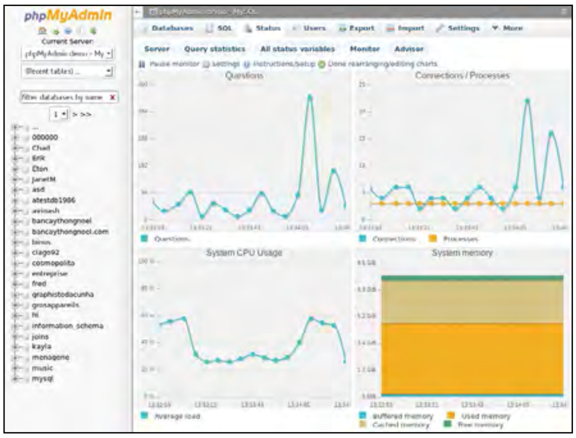
Tux on

Nobody likes losing data, and databases are known for housing lots of it, so performing regular back ups is a good idea. We can use the **mysqldump** client to do this. Rather than saving the data as stored by the engine, **mysqldump** will, by default, output the SQL commands necessary to recreate the databases and tables, so that they can be imported by some other relational database management system if needs be. It can also output CSV or XML data. To save our simple **linuxes** table, do:

```
$ mysqldump lxfdata linuxes -u root -p
```

You need to do this as *MariaDB*'s root user since tables are locked prior to dumping, and this requires privileges beyond those assigned to our humble **lxfuser**. Hopefully that works and you will see some messy looking SQL. Of course, it's more convenient to redirect this to a file, by adding something like **> linuxes.sql** to the command above. You could then import the data like this:

```
$ mysql lxfdata < linuxes.sql
```

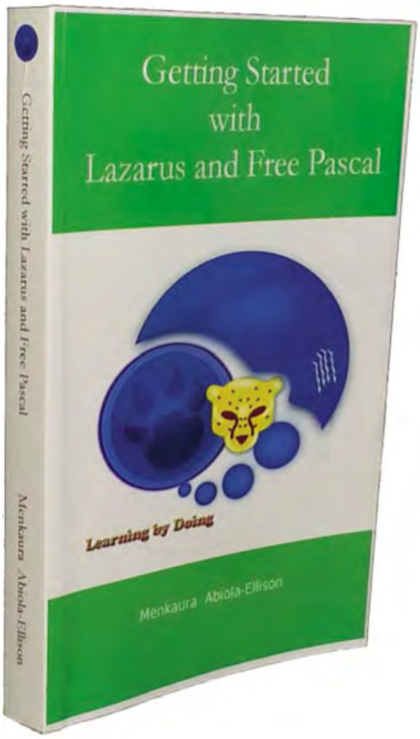


› ***PHPMyAdmin* is a popular tool for web-based database administration. It's also a popular target for hackers, so be careful.**

Note that this will replace, rather than add to, the extant **linuxes** table, but it will not recreate the whole **lxfdata** database in which it is housed. That functionality can be achieved by omitting the table name from the **mysqldump** command. Indeed, if you want to back up all the databases (except the internal information schema one) you can omit the database name too. For more details on **mysqldump** check **https://mariadb.com/kb/en/mariadb/mysqldump**.

And that concludes our primer on the wholesome SQL database that is *Maria*. Besides the basic SQL we've shown here, you can use it in conjunction with any number of popular applications, including *OwnCloud*, *Wordpress* and *Drupal*. So don't delay, snub Larry today. LXF

Next issue: Dive into Haskell



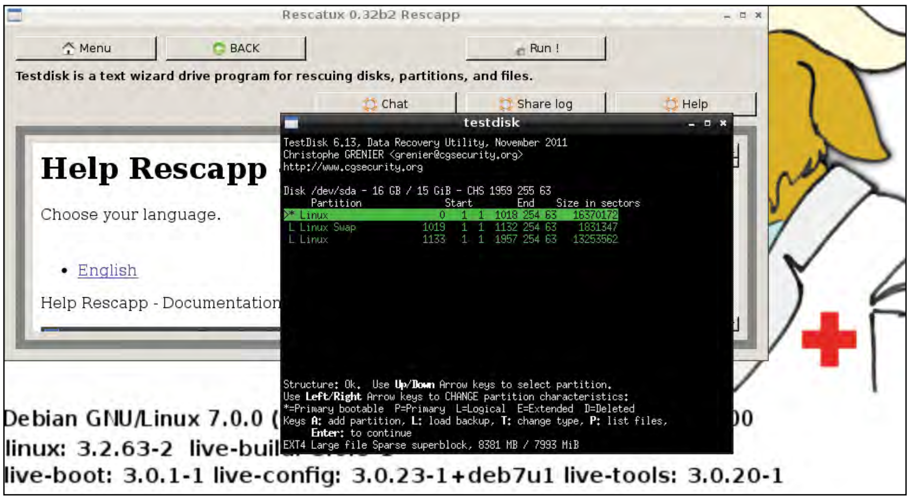
Answers

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 GPT go bye-bye

Q **I have an Acer TMP253/M with Windows 7 Professional 64-bit. I've loaded Linux onto my own and many family and friends units over the past 10 years or so, but this is the first one with the latest BIOS. I bought and followed your Beginners Guide, and with fast boot disabled, as per your magazine, I booted from the LXF192 DVD into Ubuntu 14.10. As no Windows install was found I backtracked as stated and followed the instructions on page 15, running sudo gdisk dev/sda. On rebooting into the LXFDVD I still found no Windows or any other partition, and on restarting the computer no Windows was found either, being informed that no OS was loaded and to insert a bootable disk, so I am now only able to boot from the DVD, and don't have a bootable Windows disc at the moment. The only thing wiped was GPT, as per the instructions, so I have no idea where to go to get back to correcting the booting into Windows. Your tutorial on Rescatux looks as if I could go down that route, but before I go further I need to be sure, so can you help me to restore the boot to Windows?**
G. C. Green



Testdisk is a text wizard drive program for rescuing disks, partitions, and files.

Help Rescapp

Choose your language.

- English

Help Rescapp - Documentation

```
TestDisk 6.13, Data Recovery Utility, November 2011
Christophe GRENIER <grenier@cgsecurity.org>
http://www.cgsecurity.org

Disk /dev/sda - 16 GB / 15 GiB - CHS 1959 255 63
     Partition               Start        End    Size in sectors
>* Linux                     0   1  1  1018 254 63   16370172
 L Linux Swap             1019   1  1  1132 254 63    1831347
 L Linux                  1133   1  1  1957 254 63   13253562




Structure: Ok.  Use Up/Down Arrow keys to select partition.
Use Left/Right Arrow keys to CHANGE partition characteristics:
*=Primary bootable  P=Primary  L=Logical  E=Extended  D=Deleted
Keys A: add partition, L: load backup, T: change type, P: list files,
     Enter: to continue
EXT4 Large file Sparse superblock, 8381 MB / 7993 MiB
```

```
Debian GNU/Linux 7.0.0 (                                        00
linux: 3.2.63-2  live-buil
live-boot: 3.0.1-1 live-config: 3.0.23-1+deb7u1 live-tools: 3.0.20-1
```

› ***Testdisk*, on Rescatux will rebuild your partition table after an accident.**

A Running **gdisk** will have let you delete the partition table, so nothing could be found, and Windows would be unable to boot. However, unless you've written anything else to the disk, the filesystems are still there; it's only the table that shows where they are that's missing.

Fortunately, there is a tool that will scan your disk for partitions and recreate the partition table, and it's called *Testdisk*. This needs to be run from a live CD, such as Rescatux. Scroll down to the bottom of the Rescapp window to find *Testdisk* in the Expert Tools section and run it. Select the option to create a log file, this may be useful if things go wrong, then select your disk – usually **/dev/sda** (**/dev/sr0** will be your DVD drive). Next, select the type of partition table, which should be EFI GPT for you, older systems using an MBR partition table would use the Intel option. You will then see the current partition structure, which will probably be empty as you have deleted the partition table, so select the Quick Search option. Despite the name, this may take a little while, so don't be concerned if your computer is unresponsive.

Once the scan is complete, *Testdisk* will show you what it found. Press Enter to move to

Enter our competition

Win!

if ($letter == winner) get $books

Get your questions answered and exploit our generosity.

***Linux Format* is proud to produce the biggest and best magazine about Linux and free software that we can. A rough word count of LXF193 showed it had 55,242 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 more than most of our competitors, and as for the best, well… that's a subjective claim, but it's one we're happy to stand by.

As we like giving nice things to our wonderful readers, the Star Question each month will win a copy of Martin O'Hanlon's *Adventures in Minecraft*, and it's a great intro to fun Minecraft API projects for you or a youngster. For a chance to win, email a question to **lxf.answers@futurenet.com**, or post it to **www.linuxformat.com/forums** to seek help from our lively community of readers.

› See page 94 for our star question.

the next screen. If you are happy with what it found, select Write to save the partition table back to the disk, otherwise select Deeper Search for a more extensive scan of your disk. Finally, select Quit, reboot from the power button on the Rescatux toolbar and then it's a case of crossing your fingers!

It is easy to say this after the event, but if you have anything of value on your hard disk, you should ensure it is backed up somewhere safe. This is only one of many ways in which hard disk data is vulnerable.

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**.

2 We sed it

Q In your excellent article on improving performance of your Linux box, [Speed Up Linux, p34, LXF188], line 5 of the following code supplied comes up with an error:

```
sudo sed --in-place 's/NoDisplay=true/
Nodisplay=false/g' *.desktop
sed:-e expression#1,char35: unknown option
to `s
```

As a newcomer I don't know enough about *sed* to work out how this command works and cannot, therefore, spot whether there's a typo in the magazine or that I have a problem with my typing and spaces etc.

rlcoop

From the forums

A The replace string works here, although you probably don't want to change **NoDisplay** to **Nodisplay**, the case usually matters. To explain what the command does, *sed* takes a script and applies it to the input. In this case, the script is

```
s/NoDisplay=true/Nodisplay=false/g
```

The **s** at the start tells *sed* to perform a search and replace, the part between the first two slashes is the text to replace, the next part is the replacement text. The slash at the end is required, the following **g** stands for global, telling *sed* to replace all occurrences and without **g** it only replaces the first instance on each line. Normally *sed* reads from a pipe or the files given on the command line and prints its output, suitable for redirecting to a file or piping to another program. The **--in-place** option tells it to modify the file in place instead (it actually writes its output to a temporary file and then replaces the original with that file). You don't say which distro you are using, or how old it is, but older versions of *sed* don't accept multiple input files in conjunction with **--in-place** so you may need to run it separately for each **.desktop** file. Run it for one and, if that gives no errors, you can do them all with a short shell loop:

```
for i in *.desktop; do
  sed -i 's/NoDisplay=true/NoDisplay=false/g'
"$i"
  done
```

This runs the command once for each matching file, the quotes around the file marker are there in case file names contain a space. You can do a similar thing with the **find**:

```
find -name '*.desktop' -exec sed 's/
NoDisplay=true/NoDisplay=false/g' "{}" \;
```

Surprisingly enough, **find** finds all files matching the pattern and then runs the **exec** command once for each, replacing **{}** with the file name. Note: I'm use of quotes again, just in case.

3 Non-booting DVD

Q In my quest to rid myself of the clutches of the dreaded Windows Vista, I downloaded Mint 17 and burned it to disk. Alas, it failed to boot! Taking it over to my FOSS machine running Mint 13 (I know, I know, I should upgrade) and running md5sum I got the required affirmation that the download went well. Then, applying sudo fdisk -l distro.iso (as per [Your Linux Problems, Solved p40, LXF189], I got the information that there was a boot track, but despite the size, it was marked as 'empty'. Am I missing something?

```
Device Boot
Start        End      Blocks   Id  System
/media/20150112_155405/linuxmint-17.1-
cinnamon-64bit.iso1   *          0     3026591
1513296    0  Empty
/media/20150112_155405/linuxmint-17.1-
cinnamon-64bit.iso2          2958600     2963143
2272   ef  EFI (FAT-12/16/32)
```

John Heselton

A The ISO works fine here, both when burned to a DVD and used to boot a virtual machine. The advice about running **fdisk** on an ISO image only applies if you want to use it on a USB stick. The Empty designation is harmless, it merely means that the partition does not have a filesystem ID. As you can see from the start and end values, it is anything but empty.

Did you try booting the disc in your "FOSS machine"? It has Mint installed already, so it's presumably capable of booting from a DVD. As the ISO's checksum is correct the most likely causes of this problem are either: a setting in the computer's BIOS or EFI firmware preventing booting from DVD, or a faulty DVD drive or a bad burn of the ISO. The first two

»

A quick reference to...

Tab completion

There are a number of reasons why Linux users may avoid using the command line. Remembering the various commands may be one; there's also the feeling that typing in long commands and filenames is slower than a quick mouse click on an icon. And, if you are a lousy typist like me, there's always the risk of it not working because you typed something incorrectly.

The good, or bad depending on your perspective, news is that none of those reasons are really valid, thanks to one of the shells' most user-friendly features: tab completion. What is tab completion? The easiest way to explain is with an example. Let's say you want to read the file **/usr/share/sane/xsane/doc/sane-xsane-fax-doc.html** (this file actually exists on my computer), you could fire up a browser and try to type the path with no errors, or you could use a shell and type

```
fire[TAB]/us[TAB]sha[TAB]sa[TAB]x[TAB]
d[TAB]sa[TAB]x[TAB]f[TAB]
```

The first tab searches the command path for a matching command (**firefox**) and subsequent presses of tab try to complete a filename relative to the current directory. Not only is this much quicker than typing the full name, it avoids any errors since it can only match on files that exist.

What happens if more than one command or file matches? The shell will complete as much as possible, if you press the tab key again, it will show a list of possible matches. You can then add one or two more characters and press tab again. This way, you can also use tab completion to see what commands are available. Used properly, it can greatly speed up and enhance command line working. Some shells, such as *Bash* and *Zsh*, can be extended to use completion for other things, such as hostnames and program arguments.

» causes can be eliminated if the system boots from other DVDs. The bad burn may be due to a poor quality disc, especially if it's a rewritable disc, which don't work as well in some drives as pure recordables, because they are less reflective. Try with a new DVD-R, it may also help if you lower the burning speed, perhaps to 2x. You don't say how you are writing the ISO to a disc, but all of the popular CD/DVD burning programs have options to write an ISO file to a disc, and generally offer the option to restrict the writing speed.

Depending on the computer you are trying to boot from the DVD, some require a certain amount of hoop-jumping when booting from anything but the primary hard drive try the F11 key. Incidentally, you can quickly test boot an ISO image with the Qemu emulator. Install the **qemu-kvm** package to get it. Use one of these commands, the former is faster, only use the second if this doesn't work.

```
kvm -boot d -cdrom linuxmint-17.1-cinnamon-64bit.iso
qemu-system-x86_64 -boot d -cdrom linuxmint-17.1-cinnamon-64bit.iso
```

You can do this with *VirtualBox*, which also works on Windows, but the kvm one liner is a much faster method.

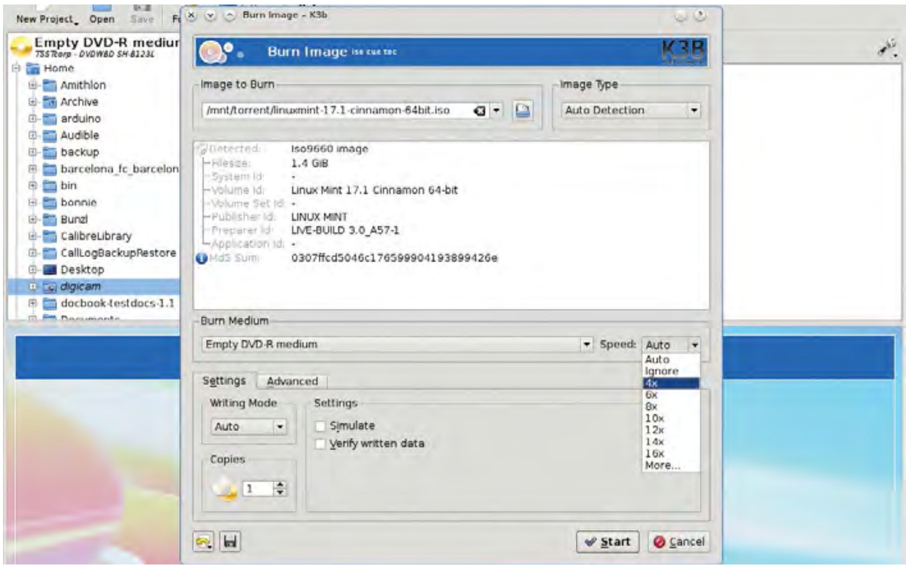


Image to Burn: /mnt/torrent/linuxmint-17.1-cinnamon-64bit.iso
Image Type: Auto Detection

Detected: Iso9660 image
Filesize: 1.4 GiB
System Id: -
Volume Id: Linux Mint 17.1 Cinnamon 64-bit
Volume Set Id: -
Publisher Id: LINUX MINT
Preparer Id: LIVE-BUILD 3.0_A57-1
Application Id: -
Md5 Sum: 0307ffcd5046c176599904193899426e

Burn Medium: Empty DVD-R medium Speed: Auto

Settings | Advanced
Writing Mode: Auto
Settings: □ Simulate □ Verify written data
Copies: 1

› When writing bootable DVDs, use a new DVD-R or DVD+R disc rather than a rewritable.

4 Windows bootloader

Q So, since I haven't really been playing around with Linux for a while, I have decided to put Ubuntu on my second hard drive. However, this hard drive has a NTFS partition already on it, and I want to keep the default Windows boot loader, if possible. Since I know this is going to require a custom install, and I don't want to hose Windows, could you point me in the direction how to do this?
Alexander Sarosi

A If you have a UEFI system, there's no issue as UEFI itself is the bootloader and you only need to load a boot manager, which you do by holding down the relevant Function key or Esc when you boot. This is because UEFI allows multiple boot managers on the same disk. In that case, just let Ubuntu do its thing. If you have a BIOS/MBR system, then you are stuck with a single bootloader and if you want to keep the Windows one, things are not as easy as they used to be. Previous Ubuntu releases had an alternate install image, which gave you the option to install the Ubuntu bootloader to the root partition rather than the MBR. If you want that you need to install Lubuntu and then install Unity on top of it. There is an alternate Lubuntu installer.

This will leave your Windows bootloader untouched, but also leave you no option to boot into your new Ubuntu system. To do that, you need to add an entry to the Windows boot menu. The easiest way to do this is with the

Star Question Winner! This month's winner is John Sorkin. Get in touch with us to claim your glittering prize!

★ Inaccessible server

Q I'm trying to run a server on my LAN which will give me access to *RStudio* on my iPad when I'm away from my home. I can connect to my server when I'm within my home LAN, but can't access the server from the Internet. It connects to a wireless router which connects to the internet via a cable TV modem.

I am running *RStudio* server on my Linux Mint 14.1 box. When I try to access the server on my LAN I have no problem: http://192.168.0.108:8787 connects to *RStudio* server. When I try to reach it from the web, I can't: http://73.213.144.65:8787 leads to "This web page is not available". I got my IP address by going to one of the websites that reports your IP and by querying my wireless router. Can you help me understand what I need to do please? I made IP address 192.168.0.108 static. I set port forwarding for 8787. I turned off UPnP.
John Sorkin

A This problem is usually caused by not setting up port forwarding properly, but you appear to have covered that. If you're sure that the external IP address you are using is correct, ISPs use dynamic addressing so can change your external address without notice at any time, that leaves the possibility that your ISP is blocking the access. This is quite common, they don't want people running servers on their networks or it can be done in the name of security. Your first step should be to check with your ISP, it may be that access is allowed on some ports, in which case just forward that incoming port on your router to port 8787 on your server, or run the server on the allowed port.

If all incoming traffic is blocked, you can still do what you want but it requires you to have access to another computer outside of your network that does allow incoming connections. Then you can use the magic of SSH reverse tunnelling. The only requirement is that it's running Linux (although this is possible with other OSes too) with *sshd* running, and that **/etc/ssh/sshd-config** contains the line:

```
GatewayPorts yes
```

This other computer must have a static address or domain name (you can use a dynamic DNS service if necessary). Now you open the tunnel on your server by running:

```
ssh -R 8000:localhost:8787 user@othercomputer
```

Because this command is run on your server, it's an outgoing connection and will not be affected by any blocks on incoming ports. Now, when you connect to **http://othercomputer:8000** all traffic will be forwarded to port 8787 on your server. You don't need to use different port numbers, it just makes the example easier to understand. There's no longer any need for port forwarding on your router. This does require an external computer, but VPS (Virtual Private Server) packages are available for a few dollars a month and free you from the shackles of your ISP's firewall.

Windows program *EasyBCD* (**http://neosmart.net/EasyBCD**). Boot into Windows, install *EasyBCD* and run it. Click on Add Entry and select *Grub 2* from the options. Click on the Write MBR button and you should now have a Windows boot menu that includes the *Grub* option for Ubuntu.

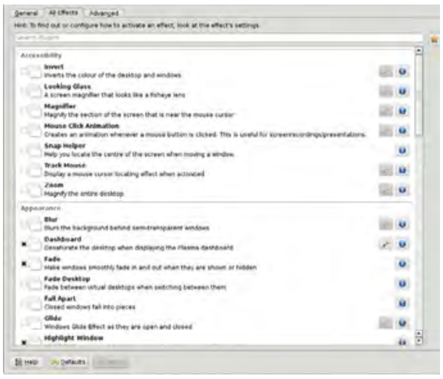
It would be simpler to allow the Ubuntu installer to take the default route of installing *Grub* to the MBR. Should you subsequently decide you want to go back to Windows, you can use *EasyBCD* to reinstall the Windows bootloader, just make sure you do this before you delete the Ubuntu partitions so that you can still boot into Windows to run EasyBCD.

5 Hidden display

Q I downloaded Mint 17.1 KDE 64-bit and installed it on a system that was originally running Mint 16 64-bit Debian fine. The only problem I had was when I ran videos in YouTube and Facebook, it was slow. I only used Mint 16 for about a week, and I wanted to try it out over the Ubuntu I had as I didn't like the Unity desktop. With Mint 17.1, I've a more serious problem. The screen desktop doesn't show what's there. I have to hover over the pop-up window to see what's in it. Also when I click on the Start menu. I don't see the items unless I hover over it. I've attached the suggested output you recommend:

```
uname -a > system.txt
lspci >>system.txt
lspci -vv>>system.txt
```

Mike Racelo



Hint: To find out or configure how to activate an effect, look at the effect's settings.

Accessibility

- **Invert** Inverts the colour of the desktop and windows
- **Looking Glass** A screen magnifier that looks like a fisheye lens
- **Magnifier** Magnify the section of the screen that is near the mouse cursor
- **Mouse Click Animation** Creates an animation whenever a mouse button is clicked. This is useful for screenrecordings/presentations
- **Snap Helper** Help you locate the centre of the screen when moving a window
- **Track Mouse** Display a mouse cursor locating effect when activated
- **Zoom** Magnify the entire desktop

Appearance

- **Blur** Blurs the background behind semi-transparent windows
- ☒ **Dashboard** Desaturate the desktop when displaying the Plasma dashboard
- ☒ **Fade** Make windows smoothly fade in and out when they are shown or hidden
- **Fade Desktop** Fade between virtual desktops when switching between them
- **Fall Apart** Closed windows fall into pieces
- **Glide** Windows Glide Effect as they are open and closed
- ☒ **Highlight Window**

› 3D desktop effects and an older graphics card do not always work well together.

A I suspect the problem is caused by a combination of a fairly old Nvidia card, the nouveau drivers and KDE's desktop effects. The nouveau drivers don't have as good 3D acceleration as the closed source drivers that are supplied by Nvidia. Combine that with a relatively low-powered graphics card by today's standards, the use of KDE's 3D effects and the display could be struggling. The card and driver combination may also have had a bearing on your video playback problems. It's not that the nouveau drivers are bad – I use them myself in combination with a graphics card that's only slightly newer than yours, but without many of KDE's desktop effects turned on – but they just don't have the same 3D performance.

One way to test the performance is by running *glxgears* from a terminal window. This program renders an animation in 3D and displays the frame rate in the terminal, if you find that you don't get a reasonable and smooth performance from this, your 3D drivers are not up to scratch.

If this is the cause of your problems, you have a couple of options. You could install the Nvidia binary drivers, most distros have them in their package repositories - if not you can get them from **www.nvidia.com** (make sure you get the right version for your card). Alternatively, you can reduce the number of desktop effects in use in KDE's System Settings. You can quickly test if this is the cause by pressing Alt+Shift+F12, which disables and enables the desktop effects. LXF

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, you can use the *Hardinfo* program (**http://hardinfo.berlios.de**) 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
```

Frequently asked questions…

Virtual Private Networks

› That's a fancy sounding name, but what is it?

A Virtual Private Network (or VPN) is a network tunnelled through another network.

› I'm still impressed by the sound of it, yet still no wiser…

The main use of a VPN is to carry private network data over a public network, such as the internet. The connection runs through an encrypted tunnel, and the security of the public network can't be guaranteed, as you have no control over access to the intermediate hardware, but the encryption makes the data useless to anyone else.

› Can't you use an SSH or HTTPS connection?

Those are two types of encrypted connection used for a specific purpose: to run a shell or access web pages securely. VPN works at a lower level, so that everything is encrypted.

› Why would you need that?

Imagine you've a network in your office, locked away behind a firewall. But you need to access your work systems from home or on the road. A VPN establishes a connection between your computer and the company network, so that you're on the network as if you were sitting at your desk, but everything is encrypted before leaving the real network, making it safe to use over the internet.

› Do you need special hardware?

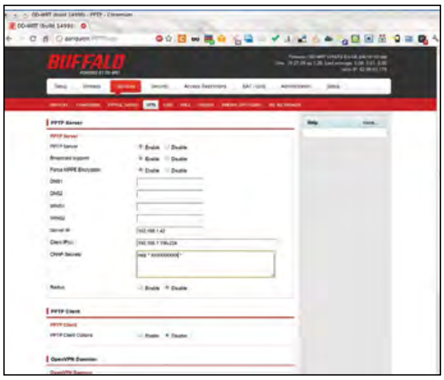
You need a VPN server running on the network, and a suitable firewall configuration, and you need VPN client software running locally – this can be running on your computer.

› That sounds expensive?

Not really, many routers support VPN, some of them costing only slightly more than the most basic non-VPN routers. Watch out for routers that advertise 'VPN pass-though'. This means they support the use of VPN over their connections, but aren't capable of managing a VPN connection and you will still need VPN client software with one of these.

› What software is needed, then?

You need both client software and a suitably configured kernel. Most distros have a suitable kernel by default. The software you need depends on the VPN variant at the other end of the link, but the OpenVPN (**http://openvpn.net**) and Openswan (**www.openswan.org**) projects cover just about all the options between them.



› You can run a VPN server on your network, but many routers include one too.

On the disc

Distros, apps, games, podcasts, miscellany and more...

The best of the internet, crammed into a phantom-zone like 4GB DVD.

Distros

What's in a name? Two of the distros on this month's *Linux Format* DVD have name issues. While Scientific Linux's name is accurate, it is also somewhat limiting in that the distro has a far wider usage. Kodi Media Center, in comparison, is a new name, because the old one had limits. Changing a name like this is a big step, while the name XBMC, which stood for X-Box Media Center, had become inaccurate it was also well known. That means that much of the work of the past few years spent in building up a reputation is at risk. Reputation and familiarity are important – we go for what we know and like. That was a factor in the rise of another of this month's distros, Linux Mint.

Is that right, though? Are we helping ourselves by sticking to our comfort zone? Do we miss out by being afraid, or at least wary of, the unfamiliar? If Linux and open source are about giving the user choice, shouldn't we embrace those choices by being more willing to try the new? Or is choosing not to do so also a valid choice? Some people go to the same holiday destination every year, others try new places. Using Linux distros is similar, which category do you fall into?

Neil

» Important NOTICE!

Defective discs

In the unlikely event of your *Linux Format* coverdisc being in any way defective, please visit our support site at **www.linuxformat.com/dvdsupport** for further assistance. If you would prefer to talk to a member of our reader support team, email us at **discsupport@futurenet.com** or telephone +44 (0) 1225 822743.

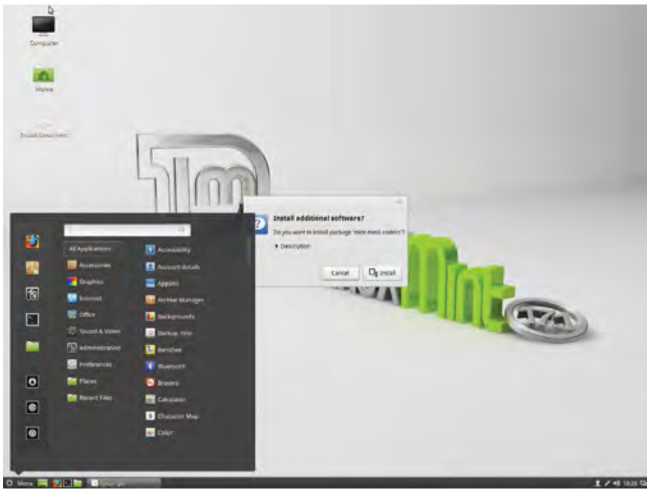
Powerful consumer distribution

64-bit

Linux Mint 17.1

Linux Mint has proved to be one of the surprise Linux distribution hits of recent years. It's success was helped initially, in no small part, by Ubuntu's decision to introduce the Unity desktop environment, which upset many that preferred a traditional desktop interface. However, Mint has continued to survive and grow it's userbase on its own merits since then, and is particularly notable for paying very careful attention to feedback from its community.

The latest release is 17.1 and we have included the Cinnamon edition on the DVD. This is version 2.4, which offers an even smoother experience from the 3D-enhanced desktop and certainly creates a good impression. Mint 17.1 is a long term release, and they really mean long term, it will be supported for five years, until 2019 – Linux Mint certainly plan to be around for a while.



Install additional software?

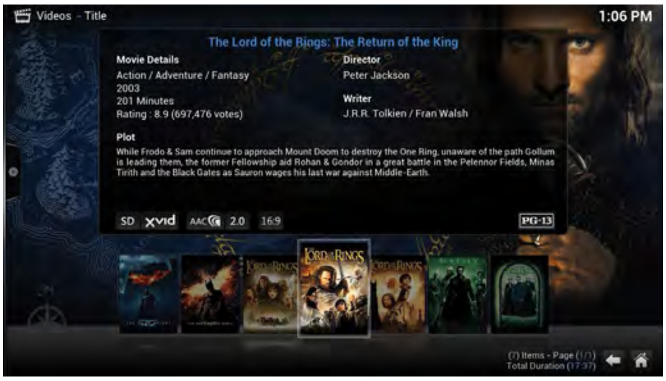
Do you want to install package 'mint-meta-codecs'?

Media centre distribution

64-bit

KodiBuntu 14.0

Some projects change names for legal reasons, some should really change their names but don't and some simply grow out of their names. XBMC started life as the X-Box Media Centre – that's the original X-Box, not the 360 or One. XMBC was ported to PCs and now that the PC has become its main platform – although the Raspberry Pi is also an option – the name has become an anachronism. So the latest version has been renamed Kodi. This is a software package that can be installed on various distros, but we have included the KodiBuntu live version, which boots straight into Kodi. You will need decent hardware to run this, and not just a 64-bit processor but also a graphics card with good 3D acceleration. If Kodi thinks your hardware is inadequate, it will dump you at a login screen but don't worry. If you then login as user **kodi** with no password and select the Lubuntu option, you get a standard desktop, from which you are able to run Kodi.



Videos - Title

1:06 PM

The Lord of the Rings: The Return of the King

Movie Details
Action / Adventure / Fantasy
2003
201 Minutes
Rating : 8.9 (697,476 votes)

Director
Peter Jackson

Writer
J.R.R. Tolkien / Fran Walsh

Plot
While Frodo & Sam continue to approach Mount Doom to destroy the One Ring, unaware of the path Gollum is leading them, the former Fellowship aid Rohan & Gondor in a great battle in the Pelennor Fields, Minas Tirith and the Black Gates as Sauron wages his last war against Middle-Earth.

SD xvid AAC 2.0 16:9 PG-13

(7) Items - Page (1/1)
Total Duration (17:37)

LINUX FORMAT DVD THE BEST DISTROS & MORE
EVERYTHING YOU NEED TO GET STARTED WITH LINUX
KODI 14.1 64-bit
The ultimate home media centre distro
MINT 17.1 64-bit
The greatest distro!
FREE eBOOK WORTH £5.99! Android
With the all-new Cinnamon desktop and long-term support for total peace of mind
LATEST PATCHES & FIXES

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

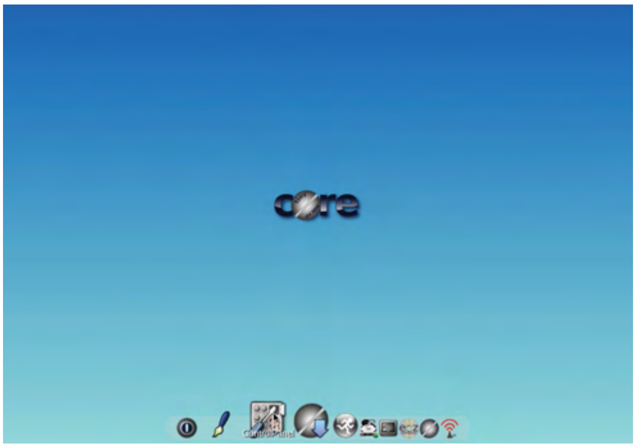
Are you reading this on a tablet?
Download your DVD from **www.linuxformat.com**

Itsy-bitsy distribution

32-bit

Tiny Core 6.0

Most of the interesting new distribution releases focus on 64-bit hardware, which is not really surprising, but there's still a lot of demand for lightweight operating systems for older hardware. Tiny Core certainly fits in the lightweight category, which has a very modular design, which uses lots of extensions that have been built by the community. We have supplied three versions on the DVD with the standard Tiny Core ISO image being only 15MB in size. There's also the Core Plus image, with many extensions built in, but at the expense of a rather bloated 75MB image. If you are hardcore and eschew graphical frippery, the smallest Core image is a mere 9MB, which still gets you a functional OS. Be warned, if you're looking for a turnkey OS with lots of pre-installed apps this isn't going to be for you.



General-purpose distribution

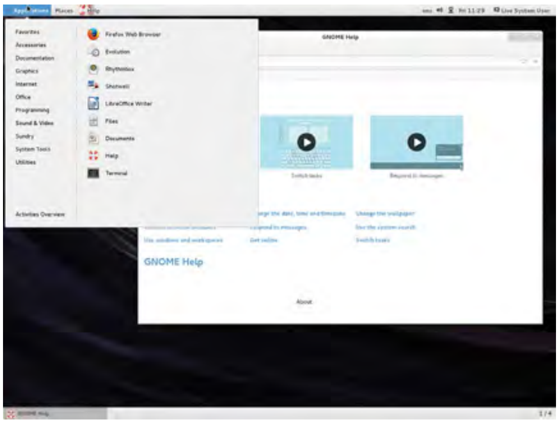
64-bit

Scientific Linux 7

This is an interesting distribution that intersects business and academia, and is useful beyond both of those fields. Scientific Linux is a respin of Red Hat Enterprise Linux (RHEL), and RHEL is a commercial product that is still completely open source, making it possible for others to build their own versions of the distro without cost (and without the RHEL support contract, of course).

Scientific Linux is sponsored by the Fermi National Accelerator Laboratory, which explains the name, but it is at its heart a general-purpose Linux distribution, the scientific tag applies more to those who developed it than to those who are using it.

In particular, if you need to gain experience of RHEL, possibly to improve your employment prospects, without purchasing a Red Hat contract, this is one way to go. If you just want to see how a distro designed for the more stable environment that's required by the commercial and academic users compares with the more fluid and cutting edge offerings available for casual desktop users, this is a good place to look.



And more!

Free eBook
Android: Your Companion Manual

This month your free 164-page eBook is all about Android, the mobile operating system that's everywhere now – it's in phones, tablets and devices that connect directly to our TVs and even slipping onto our wrists, if you're embracing the whole wearable thing.

In this guide, you'll learn the essentials from what to buy, how to get more from the built-in framework to getting organised and protecting your device. Going further, you'll learn how to automate some tasks, turn your phone into a wireless webcam, master your media and remote control music with an old phone. In all, there are 317 ways to improve your Android devices for Samsung, Motorola, HTC and more.

You'll also hear from CyanogenMod's founders, alongside a guide to installing and enhancing the best-known custom Android OS. It even covers what you can look forward to from Android reaching to 2020. And all for free. Enjoy!

System tools
Essentials

Checkinstall Install tarballs with your package manager.

GNU Core Utils 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.

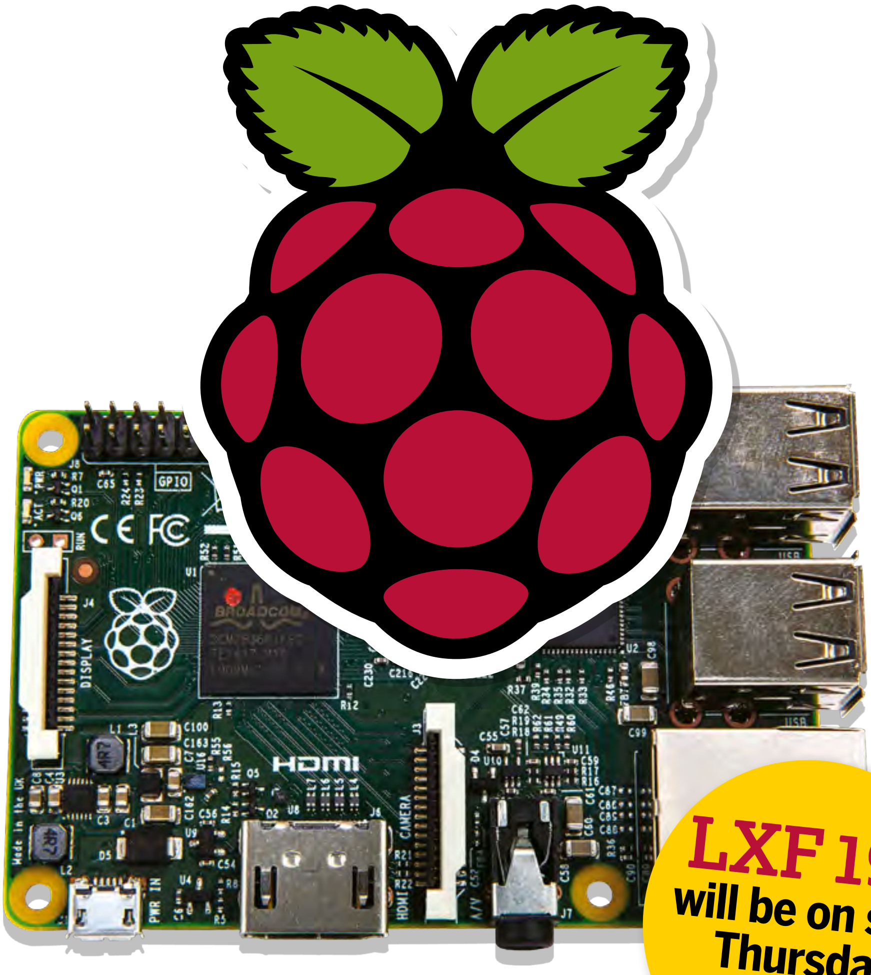
Memtest86+ Check for faulty memory.

Plop A simple manager for booting OSes, from CD, DVD and USB.

RaWrite Create boot floppy disks under MS-DOS in Windows.

SBM An OS-independent boot manager with an easy-to-use interface.

WvDial Connect with a dial-up modem.



LXF 197 will be on sale Thursday 16 April 2015

Next month: Raspberry Pi: the best new projects

Cool new projects for a cool new Pi. Put your Pi to good use streaming games, as an alarm and more!

Escape the GUI

Discover how you can do it all from the terminal! Including web browsing, video and more.

Build a Steam box

We're going Linux gaming and we're going to use Steam OS and some top gaming hardware.

Inside Apache

We delve into Apache server, so you can learn how to configure and set up every aspect of this web server.

Contents of future issues subject to change – unless we've launched ourselves into space with a Pi.

LINUX FORMAT
Get into Linux today!

Future Publishing, Quay House,
The Ambury, Bath, BA1 1UA
Tel 01225 442244
Email linuxformat@futurenet.com

abc **19,000** January – December 2014
A member of the Audit Bureau of Circulations.

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 Efrain Hernandez-Mendoza
efrain.hernandez-mendoza@futurenet.com
Editorial contributors Neil Bothwick, Chris Brown, Jolyon Brown, Sean Conway, Keith Edmunds, David Eitelbach, Kent Elchuk, Dave James, Matt Hanson, Jeremy Laird, Christopher Livingston, Conor McCormack, Juan Martinez, Les Pounder, Mayank Sharma, Shashank Sharma, Alan Stevens, Alexander Tolstoy, Mihalis Tsoukalos, Sam Tuke
Illustrations Shane Collinge, Chris Hedley

ADVERTISING
For ad enquiries please contact:
Key Accounts - sales manager Richard Hemmings
richard.hemmings@futurenet.com

MARKETING
Marketing manager Richard Stephens
richard.stephens@futurenet.com

PRODUCTION AND DISTRIBUTION
Production controller Marie Quilter
Production manager Mark Constance
Distributed by Seymour Distribution Ltd, 2 East Poultry Avenue, London EC1A 9PT **Tel** 020 7429 4000
Overseas distribution by Seymour International

LICENSING
International director Regina Erak
regina.erak@futurenet.com **Tel** +44 (0)1225 442244
Fax +44 (0)1225 732275

CIRCULATION
Trade marketing manager Juliette Winyard
Tel 07551 150 984

SUBSCRIPTIONS & BACK ISSUES
UK reader order line & enquiries 0844 848 2852
Overseas reader order line & enquiries +44 (0)1604 251045
Online enquiries www.myfavouritemagazines.co.uk
Email *linuxformat@myfavouritemagazines.co.uk*

MANAGEMENT
Content & Marketing director Nial Ferguson
Head of Content & Marketing, Technology Nick Merritt
Group editor-in-chief Paul Newman
Group art director Steve Gotobed
Editor-in-chief, Computing Brands Graham Barlow

All DVD demos and reader submissions are supplied to us on the assumption they can be incorporated into a future covermounted 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

Future is an award-winning international media group and leading digital business. We reach more than 49 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 Zillah Byng-Maddick
Non-executive chairman Peter Allen
Chief financial officer Richard Haley

Tel +44 (0)207 042 4000 (London)
Tel +44 (0)1225 442 244 (Bath)

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).

recycle
When you have finished with this magazine please recycle it.

FSC
Mixed Sources
Product group from well-managed forests and other controlled sources
Cert no. TT-COC-1815
www.fsc.org
© 1996 Forest Stewardship Council

simplifydigital
simplifydigital