FREE DVD

STORAGE DISTROS
Discover Rockstor and OpenMediaVault

NOW FEATURING LINUXVOICE

LINUX PRO MAGAZINE

FEBRUARY 2017

NETWORK STORAGE DISTROS

Who needs an expensive appliance?

Detecting Spammers with a Neural Network

OpenMediaVault: Turn your Rasp Pi into a NAS storage device

Plasma 5.8
Look inside KDE's latest desktop

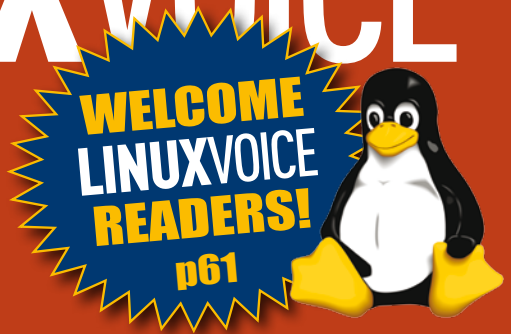
Nils Brauckmann
SUSE's CEO on the future in the cloud

GNU Social
Social networking meets Free Software

Surveillance Video Tricks
Extract action automatically with OpenCV

LINUXVOICE

- Phipps on Microsoft
- GoboLinux
- nftables
- Linux auditing tools



FOSSPicks
- Professional photo processing with Darktable
- KDE Connect: Mind meld for Linux and Android

Tutorials
- Intrusion detection
- Home storage with Nextcloud

Issue 195
Feb 2017
US$ 15.99
CAN$ 17.99

HETZNER ONLINE

Network security Firewall.

$0 Firewall free of charge

intel XEON inside

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 $128.00

monthly $ 64

* 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 $1.30 per additional TB used.

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/us

NO MISSILES...

Dear Reader,

A couple of insightful commentaries in this issue are already covering the news that Microsoft is joining the Linux Foundation (see Simon Phipps and Andrew Gregory inside). Since this column is the last to get written – usually right before we go to print, I'm really supposed to talk about things that aren't already covered in the issue, but this really is big news.

If you have been watching Linux for as long as Simon and Andrew and I have, and you remember the era when Microsoft was busily referring to Linux as a cancer, it is almost impossible not to stop and notice the change in tone. Of course, there have been lots of previous steps by Microsoft to lead the way to this announcement, such as gradually releasing the code for the .NET framework and putting Linux instances up in the Azure cloud.

Many commentators have reflected on what Microsoft becoming a platinum member of the Linux Foundation says about Microsoft – and what it says about the Linux Foundation. What strikes me is *how little* it says about either one of them – it all looks like business as usual to me. Have you looked at who *else* is a platinum member of the Linux Foundation? Other companies inhabiting the inner circle include:

- Oracle – corporate gladiator known for exerting authoritarian control over community projects. Ask the AWOL developers from the once-thriving OpenOffice and MySQL projects what they think about Oracle's commitment to FOSS principles.
- IBM – massive IT giant that wrote the book on monopolistic practices back when Bill Gates was still playing with toy trucks. IBM has done a lot for Linux through the years, but it also tops the list every year with applying for and receiving more US patents than any company in the world.
- Cisco – leader in network tech that has lots of proprietary software and megatons of proprietary hardware. They patrol the courtrooms all the time to keep competitors away from their "intellectual property."
- Intel – a major contributor to the Linux kernel (to make it work with their processors), but another corporate giant that isn't afraid to defend its near monopoly through corporate control. Do you really think their loyalty to Linux somehow trumps their legendary special relationship with Microsoft?

Other platinum members – Fujitsu, HPE, Huawei, NEC, Samsung, and Qualcomm – each have their own reasons for supporting the Linux Foundation, but they all have one thing in common: They wouldn't be doing it if it weren't good for business, and business isn't always what the free-software faithful would want it to be. Actually, if you look at the list of companies receiving new US patents for last year [1], Microsoft comes in 10th on the list, with fellow Linux Foundation sponsors IBM, Samsung, Qualcomm, and Intel all getting MORE patents than Microsoft.

So yes, the Linux Foundation should watch Microsoft and be wary of its intentions for being a platinum member, but just know that the whole platinum circle is a bunch a companies warily watching each other and warily watching the Linux Foundation, and the Linux Foundation is warily watching the other companies, too. If this all sounds sleeplessly stressful and unsettling to you, like the paranoid gold prospectors in the classic John Huston film *The Treasure of Sierra Madre* [2], that is probably why you and I are not the CEOs of gigantic corporations.

When Microsoft CEO Satya Nadella stated the new policy that Microsoft "loves" Linux, the response from the community made me think I had tuned in to a teen romance channel. "Does Microsoft really love us or is it just another line?" "Can we trust them?" "What if they change their mind and go and love somebody else?"

Seriously folks, corporations don't love – love is a chemical thing that only happens in mammals and a few species of birds. Corporations have adopted the terminology of mammalian emotional cues for reasons of convenience and self interest, but trust me on this: A corporation "loves" you the way your car loves you. You might love your car, but your car doesn't love you – it doesn't know how. *All* these platinum partners (including Microsoft) are like your car. Love really isn't the way to describe what they do. In fact, it is very likely that they would be in violation of securities laws if they based their strategic decisions on emotional attachment rather than on business interest.

So platinum membership in the Linux Foundation isn't really like being comrades in the same neighborhood street gang. It's more like being in the Security Council of the UN: "Just keep talking and keep your hands on the table. You don't launch your missiles, and I won't launch my missiles ... maybe a little spying and a few dirty tricks ... global cooperation? Sure, why not ... but hey, no missiles ... OK? ... just keep talking ... no missiles ...."

Joe Casad,
Editor in Chief

INFO

[1] List of top 10 US patent recipients: *https://en.wikipedia.org/wiki/List_of_top_United_States_patent_recipients*

[2] *The Treasure of Sierra Madre*: *https://en.wikipedia.org/wiki/The_Treasure_of_the_Sierra_Madre_%28film%29*

LINUX MAGAZINE

WHAT'S INSIDE

This month we show you how to set up a network-attached storage system with a Raspberry Pi and explore the latest release of KDE's Plasma Desktop.

Other highlights:

- **Spam-Detecting Neural Network** – Create a home-built neural network using Google's TensorFlow library, and train it to look for spammers (page 37).
- **Image Recognition** – The Perlmeister extracts action sequences from surveillance videos using the OpenCV image recognition tool (page 52).

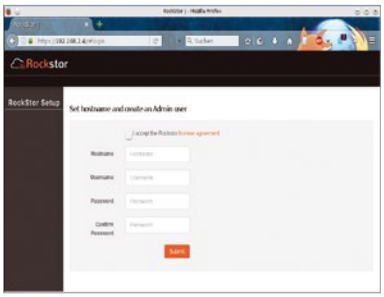
Also, lots more at Linux Voice, including a look at the innovative GoboLinux and a study of Linux auditing tools.

SERVICE

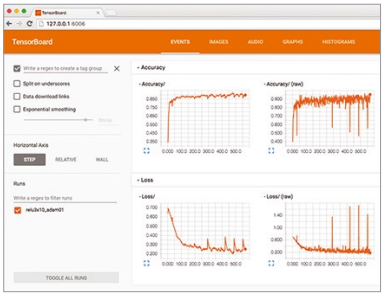
NEWS

REVIEWS

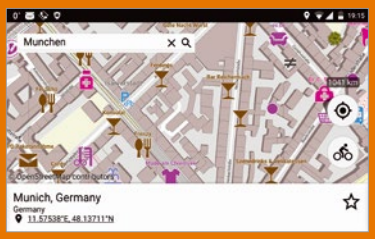
COVER STORIES



IN-DEPTH



LINUXVOICE



On the DVD

openSUSE Leap 42.2 (64-bit Install)

On this DVD you'll find the second version of the Leap 42 series, which shares its codebase with SUSE Linux Enterprise 12 (Service Pack 2). Well tested and highly stable, Leap 42.2 installs with either the KDE Plasma or Gnome desktop and comes with a back-ported Data Plane Development Kit and Open vSwitch. Xen is now supported by default in the kernel. Leap 42.2 receives packages, maintenance, and bug fixes from the openSUSE community and SUSE engineers. Starting from version 42.1, the series comes with a minimum of 36 months of maintenance and security updates.

Fedora 25 Workstation (64-bit Live)

Gnome is the default environment for Fedora Workstation, and the newest version features Gnome 3.22. You'll find multifile renaming and integrated compressed file capabilities in the Files app, a redesigned keyboard settings tool, a new landing page and easier category browsing in the Software app, and variable-speed playback in the Videos app. Also in Fedora 25 Workstation:

- Wayland display server
- Fedora Media Writer
- MP3 decoding support
- Flatpak support

ADDITIONAL RESOURCES

[1] openSUSE Leap 42.2: *https://en.opensuse.org/Portal:42.2*

[2] Leap 42.2 release notes: *https://doc.opensuse.org/release-notes/x86_64/openSUSE/Leap/42.2/*

[3] Fedora 25 Workstation: *https://getfedora.org/en/workstation/*

[4] What's New in Fedora 25: *https://fedoramagazine.org/whats-new-fedora-25-workstation/*

Defective discs will be replaced. Please send an email to subs@linux-magazine.com.

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**

SERVERS · WEB HOSTING · DOMAIN NAMES · EXCHANGE EMAIL

fasthosts

NEWS

Updates on technologies, trends, and tools

THIS MONTH'S NEWS

Biggest Kernel Release Ever

Linus Torvalds announced the release of Linux kernel 4.9 stating that it's the biggest release ever. Torvalds wrote on the LKML mailing list, "I'm pretty sure this is the biggest release we've ever had, at least in number of commits."

Linux 4.9 comes with more than 22 million lines of code. "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 reorganizations 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.)," said Torvalds, "In contrast, 4.9 is just big."

Image © iqoncept, 123RF.com

Some of the most interesting features of Linux 4.9 include support for the $5 Raspberry Pi Zero device. The release also comes with support for the Greybus driver subsystem, which was developed by Google for the now defunct modular phone concept Project Ara.

With Linux 4.9 out of way, the merge window for 4.10 is now open, but because of the holiday season, Torvalds warned developers to send their patches soon, because he will stop pulling on 23 December, and if he got "roped into Xmas food prep, even that date might be questionable," said Torvalds. "I suspect we all want a nice calm winter break, so if your stuff isn't ready to be merged early, the solution is to just not merge it yet at all, and wait for 4.11."

CentOS Linux 7 (1611) Released

Red Hat Enterprise Linux clone CentOS has announced the release of version 7. The latest version of this community-maintained distribution is based on RHEL 7.3, which was released recently.

Karanbir Singh, the release manager of CentOS wrote on the mailing list, "As with all CentOS Linux 7 components, this release was built from sources hosted at git.centos.org. In addition, SRPMs that are a by product of the build (and also considered critical in the code and buildsys process) are being published to match every binary RPM we release. Sources will be available from *vault.centos.org* in their own dedicated directories to match the corresponding binary RPMs."

CentOS is one of the most popular community-based distributions on web hosting services that competes with DEB-based distributions like Debian and Ubuntu. In 2014, Red Hat acquired CentOS, while keeping it an independent project.

Singh wrote on the mailing list, "This release supersedes all previously released content for CentOS Linux 7, and therefore we highly encourage all users to upgrade their machines. Information on different upgrade strategies and how to handle stale content is included in the Release Notes."

openSUSE Rolls Out 42.2 Leap

The openSUSE community has released the latest version of their desktop operating system, openSUSE 42.2 Leap. The latest release is a minor upgrade to the 42.x branch that comes with many new features.

Douglas DeMaio, of the openSUSE release team, wrote, "openSUSE Leap 42.2 is powered by the Linux 4.4 Long-Term-Support (LTS) kernel and is a secure, stable, and reliable server operating system for deploying IT services in physical, virtual, or cloud environments."

openSUSE Leap 42.2 is based on SLE SP2, which was released a few weeks ago. According to DeMaio, Leap 42.2 gets some of its source code from SLE 12 Service Pack 2. "New technologies such as NVDIMM, OmniPATH, and the Data Plane Development Kit with Open vSwitch are backported for the release. XEN no longer requires its own kernel and is supported by the default kernel. Along with the shared SLE codebase, openSUSE Leap 42.2 gets packages, maintenance, and bug fixes from the openSUSE community and SUSE engineers. The 42 series of Leap achieves at a minimum 36 months of maintenance and security updates starting from 42.1."

openSUSE 42.2 comes with KDE's Plasma 5.8, the LTS version of Plasma that was created specifically for openSUSE. It also comes with Gnome 3.20.2, a bit older version of Gnome. openSUSE Leap is not positioned as a cutting edge distribution and is the most stable openSUSE experience, which means the developers are a bit conservative when it comes to packages. Packages go into Leap once they are fully tested. If you are looking for the latest packages, try the Tumbleweed rolling release edition.

In 2015, openSUSE moved the openSUSE base to SUSE Linux Enterprise (SLE) SP1, which brought the two distributions closer to each other. The openSUSE community also announced a rolling release distribution called Tumbleweed, in honor of Linux kernel developer Greg Kroah-Hartman. Tumbleweed is now the upstream for openSUSE Leap and, in part, for SUSE Linux Enterprise. Every package that goes into openSUSE Leap or SUSE Linux Enterprise has to go through Tumbleweed, although enterprise customers can also work directly with SUSE Linux Enterprise teams to get those features in SLE directly.

openSUSE Leap 42.2 is available now for free download: *https://software.opensuse.org/developer/*.

Cutting-Edge Fedora 25 Released

The Fedora Project has announced the release of Fedora 25, the latest version of their fully open source Linux-based operating system. Fedora is known as a cutting edge Linux distribution, and this release comes with some of the latest open source technologies.

"The Fedora operating system seeks to deliver the latest innovations in the world of free and open source software to our users, from next-generation display servers to powerful application development tools," said Matthew Miller, Fedora Project Leader. "Fedora 25 helps to achieve this goal with the long-awaited debut of Wayland, the addition of a streamlined upgrade path, and a new edition designed to take advantage of Linux containers."

Fedora Workstation is targeted at developers and power users, although it is also suitable for average PC users. It's a preferred Linux distribution for many DevOps users, and the choice of packages that comes installed on the Workstation version is evidence of that fact. Fedora 25 Workstation comes with the latest version of Docker 1.12; Node.js 6.5, the latest version of the popular server-side JavaScript engine; multiple Python versions (2.6, 2.7, 3.3, 3.4, and 3.5) to help test across multiple Python configurations; and support for Rust, a programming language that aims to make development faster and more stable.

On the desktop side, Fedora 25 Workstation comes with Wayland, replacing the aging X11 system. Gnome 3.22 is the default desktop environment that includes

MORE ONLINE

Linux Magazine

www.linux-magazine.com

Off the Beat • Bruce Byfield

The Obstacles to Linux Security
Improving security and privacy is the most important issue in modern computing. Yet even Linux, whose architecture gives it a built-in advantage, is moving slowly on these issues. The prevailing attitudes and the innate difficulties of bringing security to the desktop mean that the progress is slow.

Ethical Boundaries and Free Software in the Reign of Trump
How far would you compromise your ethics? In the aftermath of the election, this question is suddenly ruthlessly practical for Americans.

Phones are the New Average
A few weeks ago, I traded in my phone. I prepared the way I usually prepare when buying hardware, looking up the specs, and making a spreadsheet for comparisons, but the task didn't motivate me. When I realized I was avoiding the task, I took my spreadsheet down to the nearest mall kiosk and listened to the clerk's description of several of the most popular phones. But somewhere in the middle of the descriptions, I realized I had stopped listening, and bought one at random.

ADMIN HPC

http://hpc.admin-magazine.com/

Diving In • Thorsten Scherf
GlusterFS stores data across the network and can be used as a storage back end in cloud environments.

ADMIN Online

http://www.admin-magazine.com/

Exploring OpenStack's Trove DBaaS
Martin Loschwitz
DBaaS moves the database service to the cloud, promising a new database instance at the click of a mouse.

Proactive Monitoring • Dirk Röder
System administrators usually take action after monitoring software indicates the failure of a service or server. In contrast to this reactive approach, a proactive monitoring solution with Riemann allows admins to detect problems in advance.

Countering Embedded Malware Attacks
Thomas Gronenwald
With the resurgence of sophisticated macro virus attacks, new countermeasures are in order. We offer a few recommendations.

much-awaited features like batch file renaming, a redesigned keyboard settings tool, and additional user interface improvements. Workstation users will also be pleased with the inclusion of decoding support for the MP3 media format. In addition to these changes, Fedora 25 comes with a wide range of pre-installed desktop applications.

Fedora comes in three versions: Workstation, Atomic Host (previously Cloud), and Server. All versions are available for free download. For more information on obtaining Fedora 25, see the Fedora website.

Say it Ain't So! Microsoft Joins the Linux Foundation

Microsoft has joined the Linux Foundation as a platinum member. Microsoft made the announcement today at the Microsoft Connect 2016 event in New York.

"The Linux Foundation is home not only to Linux, but many of the community's most innovative open source projects," said Scott Guthrie, Executive Vice President, Microsoft Cloud and Enterprise Group. "We are excited to join The Linux Foundation and partner with the community to help developers capitalize on the shift to intelligent cloud and mobile experiences."

Microsoft has also released the public preview of SQL Server for Linux, which allows customers to test SQL Server on Linux and Linux-based Docker containers.

Microsoft is also partnering with Samsung to introduce their Visual Studio Tools for Tizen, a Linux-based operating system that is hosted by the Linux Foundation. The tool allows developers to build .NET apps for the Tizen operating system, which runs on millions of devices, including TVs, wearables, mobile devices, and many IoT devices.

Despite its long-standing reputation as the archenemy of Linux, Microsoft has emerged in recent years as one of the leading contributors to open source projects; their contributions on GitHub are evidence of the fact that the company is investing heavily in Linux and open source technologies. Microsoft has developed an operating system for networking switches in Azure that runs on the Linux kernel and has released many of its core products as open source, including .NET and PowerShell.

Image © bowie15, 123RF.com

How to Bypass Authentication on a Linux System

Researchers have discovered a flaw in the Cryptsetup utility that allows an attacker to bypass the authentication process on some Linux-based systems just by pressing and holding the Enter key for 70 seconds.

Debian/Ubuntu-based systems with encrypted system partitions are affected by this vulnerability. Researchers warn that other distributions using Dracut instead of initramfs are also vulnerable.

Hector Marco and Ismael Ripoll from the Cybersecurity Group explained in their security advisory that the vulnerability allows you to obtain a root initramfs shell on the affected system. "The vulnerability is very reliable because it doesn't depend on specific systems or configurations. Attackers can copy, modify, or destroy the hard disk as well as setup the network to exfiltrate data. This vulnerability is especially serious in environments like libraries, ATMs, airport machines, labs, etc., where the whole boot process is protected (password in BIOS and GRUB) and you only have a keyboard and/or a mouse," Marco and Ripol wrote.

The worst thing about this vulnerability is that you don't need physical access to the machine; it is possible to exploit the vulnerability remotely in cloud environments.

Last year, the same researchers discovered a bug in GRUB 2 that allowed an attacker to bypass all securities on a locked-down Linux machine by hitting the Enter key 28 times when asked for a username.

Zack's Kernel News

Chronicler Zack Brown reports on the latest news, views, dilemmas, and developments within the Linux kernel community.

By Zack Brown

Cgroups Xattr Security

Serge E. Hallyn said that within a user namespace (i.e., a virtual machine), a root user could not be allowed to write a `security.capability` extended attribute (xattr). If it could, then any user within that namespace could `su` to root, write the xattr, and execute the file with those security privileges on the host machine.

On the other hand, the root user on the host machine could absolutely be allowed to write a `security.capability` xattr because, of course, they're the root user. This is one of the many examples of ways in which security considerations require strange feature curtailment within virtual machines.

ZACK BROWN

The Linux kernel mailing list comprises the core of Linux development activities. Traffic volumes are immense, often reaching 10,000 messages in a week, and keeping up to date with the entire scope of development is a virtually impossible task for one person. One of the few brave souls to take on this task is **Zack Brown**.

Nonetheless, if something behaves differently on the virtual machine than on the host, that represents a fundamental incompatibility that would affect things like software portability and reliability.

Serge was working on some code that would allow the virtual machine to believe the root user had written the `security.capability` xattr, while still protecting the host system. He hoped his code "allows a simple setxattr to work, allows tar/untar to work, and allows us to tar in one namespace and untar in another while preserving the capability, without risking leaking privilege into a parent namespace."

Eric W. Biederman liked the patch but said it was strange enough to need some very careful going-over before he'd send it along to the main tree; meanwhile, Serge wrote up a man page for Michael Kerrisk, to help testers understand what was going on.

Encrypting the Running Kernel

Tom Lendacky added support for Secure Memory Encryption (SME) to the x86 architecture. With his patch in place, if the kernel were compiled with SME enabled and booted with the command-line option `mem_encrypt=on`, then the running kernel would be encrypted.

The big controversy over this patch seemed to be how to turn it on and off. Borislav Petkov pointed out that if the user could specify `mem_encrypt=on`, it would make sense to support a `mem_encrypt=off` option, in case the user preferred not to encrypt the running kernel. However, Tom replied that, absent being explicitly turned on, the encryption simply would not occur. He was fine with documenting a `mem_encrypt=off` option, though, if that would make people happier.

However, Borislav said this wasn't his point at all. He simply felt that encryption should be enabled by default, and that if a user didn't want the feature enabled, they should give a `mem_encrypt=off` command-line argument to say so.

The discussion petered out there, but presumably any kernel developer would rather have their code enabled by default, instead of having to be specified by the user.

To Run an Unreadable File

Eric W. Biederman posted some code to prevent `ptrace` from running a file that did not have read privileges set. The apparently simple idea being, if you don't have permission to read the file, then how could you have permission to run the code you couldn't read?

However, Willy Tarreau did have an objection. He was concerned that people had been utilizing this very loophole for decades to block various rootkits from taking over their systems. He did a directory listing of his own `bash`, `coreutils`, `ls`, and `telnet` tools, showing that they were marked as executable without being readable.

Willy explained that certain rootkits did their mischief by copying and modifying the code in those fundamental utilities. By making them executable but not readable, Willy said he had blocked several rootkits over the years.

Meanwhile, Willy said, Eric's patch would make it difficult to debug any code that had to be traced through any of those utilities. He said, "So here I fear that we'll break strace at many places where strace precisely matters to debug things."

As an alternative, Willy suggested that Eric make his code controllable by a `sysctl`, so it could be turned on and off as needed.

Kees Cook said, however, that once Linux began to support executable-only memory, `ptrace` would again start to fail, so trying to hang on to it as long as possible didn't make much sense. Kees suggested just biting the bullet and doing it Eric's way.

In a surprise twist, Willy's objections were enough to give Eric second thoughts. He didn't want to break historical usage, so he redid his patch to simply produce a warning instead of disallowing

running an unreadable file. Andy Lutomirski said he thought this was a better approach, too.

Reading from the Ring Buffer

Wang Nan wanted to be able to pause and resume the kernel ring buffer, to be able to read from it without worrying that anything might try to write to it at the same time. The ring buffer is where the kernel stores the log of its events. It's where the `dmesg` output originates.

Peter Zijlstra had no objection to this kind of patch, but he did want to see the man pages updated as well. And Vince Weaver agreed that updating the man pages would be good for something like this, since it represented an application binary interface (ABI) change.

ABI changes mean that user code compiled before the change might not run on kernels compiled after the change. Traditionally, ABI changes are something kernel developers desperately want to do and which Linus Torvalds absolutely refuses to allow. The reason developers want to do it is because they must otherwise support ancient legacy features forever – even broken or inconsistent features. The reason Linus refuses to allow it is because breaking the ABI means real user code starts to break in the real world. As a side issue, it becomes more difficult to find bugs in the kernel itself if the search goes across the boundary of the ABI change.

The question of why Wang's patch was important enough to justify an ABI change was not made clear during the mailing list discussion. Possibly it only added to the ABI instead of changing something that was there already. However, it does seem to be an important change, because as Wang said, "Before reading caller must ensure the ring buffer is frozen, or the reading is unreliable."

Migrating Processes Between Cgroups

John Stultz posted a patch to allow processes to migrate from one virtualized Linux instance to another. He got the idea from Michael Kerrisk, and it had originated in Android to avoid having to run the process manager with root privileges.

Typically process migration between virtual Linux instances is a risky business because it represents a potential point of attack, where hostile code might break out of the sandbox and escape to the host system.

Still, a potential security hole is different from an actual security hole, and folks like Kees Cook were glad to see the code. Andy Lutomirski, on the other hand, said that cgroups were about to expand their entire scope to do more than simply resource control. Future cgroups might have powers and abilities beyond those of mortal virtualized systems. Simply migrating a process from one cgroup to another, Andy said, might expose vulnerabilities that today's cgroups would not.

Without an idea of what to do instead, John didn't have any solid idea of how to update his patch to avoid the problems Andy was talking about. Finally, Andy suggested adding some form of privilege, not only to the task but to the cgroup itself, so that a process could only migrate from one cgroup to another if the user had permissions over both the process and the target cgroup.

Beyond that, it was a question of exactly which capabilities to use and how to organize them properly. At least the possibility does exist to support cgroup migration in the future, without compromising security.

Overall, cgroups are a strange and dangerous world. They never perfectly imitate a host system, and there is always the temptation to add bizarre features that could only exist in a virtualized environment. Ultimately, I suspect virtualized OSs will look quite a bit different from the hosts.

Performance Events Limits

Jeffrey Vander Stoep wanted to limit certain potential attack vectors, so he wrote a patch such that if `kernel.perf_event_paranoid` were set to `3`, users would have to have `CAP_SYS_ADMIN` to gain access to performance events.

Jeffrey's idea was that performance events were great for debugging purposes, but they were rarely used on production systems and represented a potential security hole. He pointed to a slew of examples and said, "This new level of restriction allows for a safe default to be set on production systems while leaving a simple means for developers to grant access."

Kees Cook was enthusiastic about the patch, but Peter Zijlstra said plainly, "We have bugs; we fix them; we don't kill complete infrastructure because of them." He went on, "the problem I have with this is that it will completely inhibit development of things like JITs that self-profile to recompile frequently used code. I would much rather have an LSM hook where the security stuff can do more fine grained control of things, allowing some apps perf usage while denying others."

Arnaldo Carvalho de Melo also pointed out other areas of development that would be stifled by "such big hammer restrictions." Daniel Micay, on the other hand, came down in support of Jeffrey's patch. He said that it would still be possible, with Jeffrey's patch, to give certain processes the privileges they needed to use performance data. He said to Peter, "You're forcing people to have common local privilege escalation and information leak vulnerabilities for something few people actually use."

Daniel added, "This patch is now a requirement for any Android devices with a security patch level above August 2016. The only thing that not merging it is going to accomplish is preventing a mainline kernel from ever being used on Android devices."

Kees also replied directly to Peter's statement, "we have bugs; we fix them." He said, "it isn't what things look like for the average end-user of Linux. The lifetime on bugs is very long, even in upstream (see both Jon Corbet and my talks about this: an average of five years from introduction to fix), and gets drawn out even further by vendors with slow (or missing) update processes. Being able to remove attack surface is a fundamental first step of security defense, and things like perf, user namespaces, and similar APIs, expose a lot of attack surface when they are enabled. And the evidence for this attack surface being a real-world risk is in the history of security vulnerabilities (that we know about!) in these various APIs."

He went on to say, "the APIs are needed, but they lack the appropriate knobs to control their availability. And this isn't just about Android: regular distro kernels (like Debian, who also uses this patch) tend to build in everything so people can use whatever they want. But for admins that want to reduce their systems' attack surface, there needs to be ways to disable things like this."

Peter agreed with the knob concept, but he felt that the specific knob being proposed was not the right one. He said, "Having this knob will completely inhibit development of such applications. Worse, it will probably render perf dead for quite a large body of developers. The moment you frame it like: perf or sekjurity, and even default to no-perf-because-sekjurity, a whole bunch of corporate IT departments will not enable this, even for their developers."

The current proposal, he said, was too coarse and inhibiting. A better way had to be found.

Kees said, "The vast majority of people running Linux do not use perf (right now). I've never suggested it be default disabled: I'm wanting to upstream the sysctl setting that is already in use on distros where the distro kernel teams have deemed this is [a] needed knob for their end-users." He pointed out, "All of the objections you're talking about assume that the knob doesn't exist, but it does already. It's just not in upstream."

Jeffrey also put in, "Far from trying to kill perf, we want (and require) perf to be available to developers on Android. All that this patch enables us to do is gate it behind developer settings – just like we do with other developer targeted features."

Ingo Molnár, however, agreed with Peter. He said that it made no difference whether the default was on or off. The coarse/limiting aspect was simply too significant and had to be dealt with properly. Ingo said, "This isn't some narrow debugging mechanism we can turn on/off globally and forget about, this is a wide scope performance measurement and event logging infrastructure that is being utilized not just by developers but by apps and runtimes as well."

He went on to say, "in practice what will happen is that if the only option is to do something drastic for sekjurity, IT departments will do it – while if there's a more flexible mechanism that does not throw out the baby with the bath water that is going to be used."

Ingo compared the current patch with a situation that might have played out in the past. He said:

This is as if 20 years ago you had submitted a patch to the early Linux TCP/IP networking code to be on/off via a global sysctl switch and told people that "in developer mode you can have networking, talk to your admin."

We'd have told you: "this switch is too coarse and limiting, please implement something better, like a list of routes which defines which IP ranges are accessible, and a privileged range of listen sockets ports and some flexible kernel side filtering mechanism to inhibit outgoing/incoming connections."

Global sysctls are way too coarse.

Daniel argued that at least with the current patch, there was a way to turn access to perf events on and off at run time. If, for example, this was a compile-time configuration option, he said, it would require a reboot to gain access to perf events.

He also said that the "wide scope" infrastructure Ingo had referred to was exactly why the security problem was so big. He said, "If it wasn't such a frequent source of vulnerabilities, it wouldn't have been disabled for unprivileged users in grsecurity, Debian, and then Android."

He reiterated that Android and Debian already included the current patch. The baby wasn't in danger of being thrown out with the bath water – it had already happened – and the official kernel could recognize that or not. He said, "They'll keep doing it whether or not this lands. If it doesn't land, it will only mean that mainline kernels aren't usable for making Android devices."

Peter suggested coming up with a new capability to govern access to perf events. Specifically, he suggested that processes operating across a network connection would drop all capabilities. This would allow perf access at the local level, but not to networked applications.

Eric W. Biederman reiterated the main objection to the current patch. He said, "the problem with a system wide off switch is what happens when you have a single application that needs to use the feature. Without care your system wide protection disappears. That is very brittle design."

Eric suggested using a sandboxing approach instead, in which a given sandbox could have a given feature turned on or off without affecting anyone else. He added, "One of the strengths of Linux is applications of features the authors of the software had not imagined. Your proposals seem to be trying to put the world [in] a tiny little box where if someone had not imagined and preapproved a use of a feature it should not happen. Let's please avoid implementing totalitarianism to avoid malicious code exploiting bugs in the kernel. I am not interested in that future."

Kees replied, saying that he was "interested in giving system owners greater control over what's exposed. That's not about limiting access everywhere. And I'm interested in making sure that the upstream kernel actually provides what end-users want. In the most extreme version of this is when distros carry kernel patches to get it done (this was true with userns and is true again here with perf). This IS a desired feature, and it exists in the world. I want to avoid the confusion that arises from people running patched kernels: upstream developers don't realize what state their features are in when they reach end users, documentation doesn't match, etc., etc."

Daniel also said, "There are perf event vulnerabilities being exploited in the wild to gain root on Android. It's not a theoretical attack vector. They're used in both malware and rooting tools. Local privilege escalation bugs in the kernel are common so there are a lot of alternatives but it's one of the major sources for vulnerabilities."

The discussion became somewhat disjointed at this point. There was some effort to explore technical alternatives to the original patch, but there was still disagreement over exactly how dangerous the various vulnerabilities were and how crucial it would be to eliminate absolutely all of them in one fell swoop. At the same time, several people who were expert in the areas they'd been discussing so far were less expert in some of the proposals that began to emerge, so various folks had to catch up to others.

Ultimately, no agreement could be reached, and the debate seemed to be shaping into one of epic proportions. The kernel loyalists are on one side, saying that a given feature would be unacceptable, and the distro makers are on the other side, saying that the feature in question are already included in systems around the world, including everything running Android or Debian.

It's impossible to know how the debate will eventually play out. These sorts of things can take years, with neither side willing to budge. In this particular case, though, it does seem there is room for a more subtle approach than the original patch would allow. ■■■

An interview with SUSE CEO Nils Brauckmann

Open Collaboration

By Swapnil Bhartiya

SUSE is the oldest Linux company, and it is still going strong. Founded in 1992, just one year after Linus Torvalds announced the birth of Linux, SUSE has gone through many changes in recent years. In 2003, Novell purchased SUSE in a bold effort to ride with the rising tide of Linux. In 2011, the Attachmate Group bought Novell, reorganizing SUSE as an independent business unit. In 2014, Attachmate became part of Micro Focus.

Nils Brauckmann came to SUSE through the Attachmate acquisition, and some believe he is the leader the company always needed. Last year, he was promoted as the chief executive officer of SUSE within Micro Focus. SUSE has been investing heavily in increasing its work force, mostly with more engineers. In this interview, we reflect on the last 25 years of Linux and SUSE.

Swapnil Bhartiya: Linux is 25 years old. What do you think it has achieved in these 25 years? What do you think it has contributed to our society that's beyond software and code?

Nils Brauckmann: The success of Linux is tightly coupled with the rise of open source innovation and the open source business model. Through massive collaboration and shared contribution by open source community members, faster innovation with reduced cost and time to market became a reality. This led to commoditization of technology, increased choice, and reduced dependencies for users and emerging technology companies.

In the past 25 years, Linux has helped achieve something truly noteworthy – making the majority of the world's shipping operating systems open source. Linux has also had tremendous influence on the very way business (especially the computer technology business) operates – for the better. The openness of Linux brings great benefits to enterprises, including involvement (if they choose) in product development and a chance for real influence on the technology that will run their business.

SB: SUSE was founded in 1992. How has the company evolved since the early days?

NB: When SUSE began as one of the true Linux and open source pioneers – we started even before the Linux kernel hit version 1.0 – we were focused on refining and providing the earliest distributions of Linux (this was before the common Linux distributions even existed). Our goal was affordable, open source Linux ready to be used by enterprise customers to run their business-critical workloads. Over time, SUSE, like Linux, has grown and evolved. We were the first company to bring Linux to mainframes, and to this day, SUSE is driving Linux usage on supercomputers. Along the way, we invested in a lot of open source projects. These days we're looking forward to a world of software-defined infrastructure, powered by SUSE Linux.

SB: SUSE has been through some rough patches in its journey, but it keeps coming back? What makes you so resilient?

NB: Great people and culture at SUSE are obvious, I think. Also, we have a commitment to the customer. We hire great engineers, and we do everything as open source. The core principle of being a truly open company has guided us well, and we are committed to engaging in dependable, trusted relationships with our alliance partners and customers.

SB: SUSE has been a Microsoft partner for a very long time. Today we see a "new" Microsoft that "loves" Linux. As a long-time partner of Microsoft, what's your perspective on this change of heart?

NB: This change is refreshing – for customers and for partners like SUSE. Microsoft, like any company, is constantly evaluating where value lies and where the opportunities to grow their business are. Our early work with Microsoft helped to demonstrate the power of Linux and open source. SUSE has been building solutions with Microsoft Azure since its launch in 2012 and collaborating with Microsoft since 2006, serving more than 1,000 joint customers. It is excellent

to see any company, Microsoft or otherwise, adopting and becoming engaged with Linux and open source.

SB: How sustainable is SUSE under the new owner Micro Focus? Can you talk about your growth?

NB: Following the acquisition by Micro Focus, the SUSE business was given a mandate to deliver "accelerated, sustainable, and profitable revenue growth" and is provided with ongoing support and investment to support this vision. This clearly shows in SUSE's business results. Fiscal year 2016 was a successful year for SUSE, with 18.2 percent growth in revenue.

To create additional capacity for ongoing growth, we also expanded SUSE headcount across different business functions and geographies and aligned the critical supporting organization much more tightly with the SUSE business.

We also extended SUSE's presence and contribution in key open source projects and industry groups. I expect this positive trend to continue in fiscal year 2017 and beyond.

SB: SUSE is also sponsor of openSUSE, what is the relationship between the product SUSE Linux Enterprise Server (SLES) and the project openSUSE?

NB: Being a truly open source company means working closely with the community. OpenSUSE, the community project and distribution, provides us with a powerful foundation to build and support SUSE Linux Enterprise, a product of SUSE, the company. SUSE does provide financial support for openSUSE, but the direction of the project is independently decided from within the project. The community defines and drives openSUSE forward. The contributors within SUSE are simply part of that community.

SB: SUSE is still a hard-core Linux company, but we are witnessing the rise of new paradigms like Docker containers, OpenStack, and Cloud Foundry. Where is SUSE in that cloud picture?

NB: SUSE is an open source company, and we embrace new open source technologies (typically running on top of Linux) that provide new functionality and new value, such as OpenStack, Docker, and Ceph. SUSE joined the OpenStack community and started contributing in 2011, and we are a founding and platinum member of the OpenStack Foundation. SUSE was the first enterprise Linux vendor to ship a supported OpenStack product. Likewise, we are a founding member of the Open Container Initiative and began shipping Docker with SUSE Linux Enterprise 12 and SUSE OpenStack Cloud in 2015.

Last year we joined the Cloud Foundry Foundation and are actively collaborating with SAP to improve the cloud provider interface between Cloud Foundry and OpenStack.

The common thread with these actions is that we see all of these efforts as being essential to helping customers move to a software-defined infrastructure. Our goal is to be the foundation on which customers can build the next-generation platform for business systems.

SB: Microsoft and Apple are releasing products as open source. AI and machine-learning companies are open sourcing their technologies. Even car companies are open sourcing stuff. Has open source won? Or are there areas where open source has yet to make a dent.

NB: Open source, as an idea, has clearly dominated significant areas of technology. Some companies still cling exclusively to closed-source software, but this seems to be changing. It's not so much that open source has "won" as it is people have been coming to the realization that open collaboration is a more productive way to approach software innovation than closed collaboration.

SB: What are the new challenges and opportunities for SUSE as we enter the world of Internet of Things (IoT), machine learning, AI, VR, and robotics?

NB: There are many new opportunities for SUSE in the context of these emerging new technology and market trends. Most of these innovations rely on Linux and other open source software as the platform, and that means SUSE is well positioned to support and capitalize on these trends. To succeed, we'll need to engage in active, strategic dialog with our customers, our partners, and our open source communities.

SB: I have heard that you are also a drummer? Are we going to see you cameo/perform in any upcoming SUSE music [parody] videos?

NB: My team has been asking, and let's say we are still in negotiations. ■■■

Build a NAS system with OpenMediaVault and a Raspberry Pi

Storage Buddy

A NAS system does not have to be large, heavy, and expensive. A Raspberry Pi and the OpenMediaVault Linux distro are a compact alternative to heavy and costly NAS. *By Erik Bärwaldt and Christoph Langner*

Network-Attached Storage (NAS) systems are often ungainly contraptions that take up lots of floor space and come with many slots for hard drives or SSDs. However, an oversized NAS solution of the conventional sort generally is not needed on a smaller network that doesn't have a large video collection or database.

Dedicated NAS devices, already costly to purchase, also hit your wallet with relatively high energy consumption. If you don't need a big bulky dedicated NAS system, you can achieve much of the same result on a small scale by rolling your own NAS solution with a Raspberry Pi (Rasp Pi).

Don't expect your Rasp Pi NAS to shoulder enterprise-level workloads. One of the issues with using the Pi as a NAS is the bottlenecks caused by its system architecture. In particular, the relatively slow Fast Ethernet interface and the mass storage, which can only be connected via USB 2.0, take a toll on performance. But if you don't have big demands for performance, a simple and unobtrusive Raspberry Pi 3 (RPi3) will work well as a NAS.

OpenMediaVault [1] (Figure 1) is a NAS-focused Linux distribution that maintains a version for the Rasp Pi. Your Rasp-Pi-based OpenMediaVault server is suitable for minor datasets such as text files or spreadsheets. At a data rate of about 9MB/s (write) and 11MB/s (read) via Ethernet, you only need close to two minutes to transfer a 1GB file. In practice, this means managing large amounts of image or multimedia data with the Rasp Pi is not much fun. However, you do not have to give up on

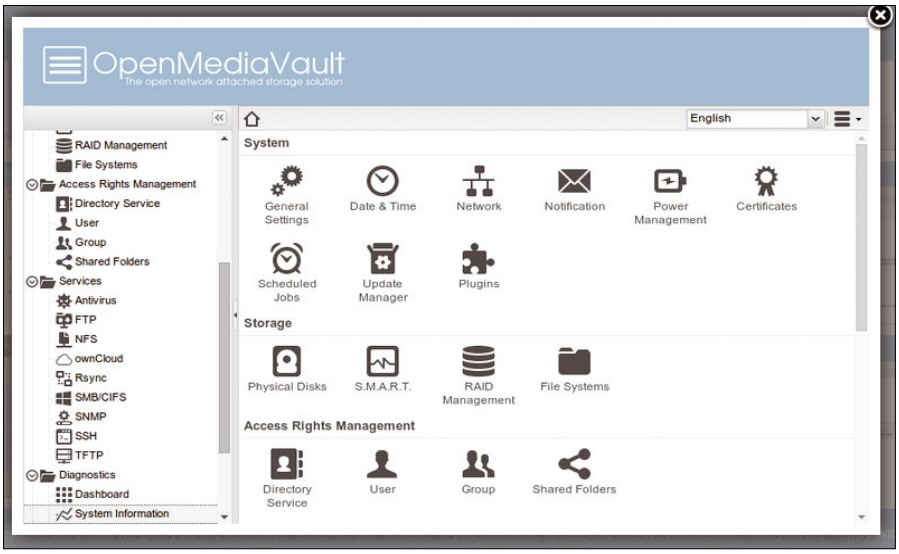


Figure 1: OpenMediaVault is a specialty Linux distro preconfigured for NAS scenarios.

OpenMediaVault if you need more performance than a Rasp Pi can offer. The developers also provide a version of OpenMediaVault for classic x86 machines.

NAS Software

Specialized operating systems are a recommended alternative to classic Linux for NAS tasks. Manual installation and configuration of all the necessary services is demanding, even for veteran Linux administrators, so a system that is already configured for NAS workloads has many benefits. Our Rasp Pi OpenMediaVault NAS has the added benefit of being tiny and unobtrusive.

As with other Rasp Pi distros, you need to burn the OpenMediaVault IMG file to a MicroSD card. Unpack the image file [2] into any directory on your computer. If you are using a Linux computer, use the `dd` command to write the image. If you are working on Windows, write the image with a graphical tool like Win32DiskImager.

As shown in Listing 1, Linux users should set the name (possibly with the path) of the IMG file for the `if=` input, and the device name of the MicroSD card in the card reader for the `of=` output. You can determine the device name with `lsblk`. Next, use `sync` to ensure that the system can empty write caches that may have been created. Then insert the card into the Rasp Pi and start it. You can configure the services via the web interface of another machine in the network.

Please take care to ensure from this point that you use a power supply unit that can provide at least 2.5A at 5V with the RPi3.

If the NAS is built with a Raspberry Pi 2 (RPi2) and an external 2.5-inch hard disk without its own power supply unit, you still need to prepare the USB ports (see the "More Power" box). However, keep in mind that not every hard disk will be a good match.

Since the RPi3 only has a Fast Ethernet interface with a maximum throughput of 100MB/s, you need to check whether the WiFi interface works faster, since it runs with a maximum of 300MB/s from the 802.11n standard onward. The nominally higher bandwidths of the WiFi interface, however, often fail less than the cable-based variety (due to the overhead during data transfer and depending on location). OpenMediaVault initially only activates the Ethernet interface, although it can cope with the RPi3's built-in WiFi interface if you configure it manually.

LISTING 1: Writing the IMG File on Linux

```
$ lsblk
NAME                    MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
sda                       8:0    0  59,6G  0 disk
|-sda1                    8:1    0  53,7G  0 part /
|-sda2                    8:2    0   5,9G  0 part [SWAP]
sde                       8:64   1   3,8G  0 disk
$ sudo dd bs=4M if=omv_2.2.5_rpi2_rpi3.img of=/dev/sde
$ sync
```

LISTING 2: Finding the IP Address

```
$ sudo apt install arp-scan
$ sudo arp-scan --localnet | grep Raspberry
```

First Run

The Rasp Pi equipped with OpenMediaVault, when connected to a screen, simply shows a text-based terminal with a login request after booting. If you wish to configure the NAS system, switch to a desktop computer in the network rather than logging into the Rasp Pi directly.

The Rasp Pi's status messages display the IP address briefly before the login screen when the Pi receives its address through DHCP. If you are working on a Rasp Pi without a monitor, you can determine the IP address from a Linux PC via `arp-scan` (Listing 2); you can use the Adafruit Raspberry Pi Finder [3] for this purpose on other systems. Alternatively,

MORE POWER

RPi2s are only set by default to supply low power via USB, meaning that many external 2.5-inch hard disks will not run. You must modify the Rasp Pi configuration in `config.txt` in order to provide more power on the USB interfaces – this modification is not necessary for RPi3. To undertake configuration work, log into the Rasp Pi OpenMediaVault via SSH as the *root* user with the password `openmediavault`. You now open the configuration file with the `nano /boot/config.txt` command and add `max_usb_current=1` to the end of it, before saving the file and restarting the Rasp Pi. An impressive 1200mA (instead of 600mA) is available on the RPi2 Model B, meaning that an external 2.5-inch hard disk will function without the need for additional power.

read the IP assigned to the Rasp Pi from the web interface of your wireless router.

The next step is to open a web browser on the desktop computer and enter the IP address of the Rasp Pi as the URL. You will then come to a login screen, from which you first set the relevant language localization by means of a selection list and then proceed to log in as *admin* using `openmediavault` as the password.

A clearly structured dashboard opens for you to call up all the configuration options from a list on the left. Next to this dashboard on the right, two small windows show the status of the individual services and simple system information.

The first step for increasing the security of the admin access is to modify the preset access data. Select the *General Settings* menu on the left before altering the login data in the *Web Administrator Password* tab. An encrypted connection via SSL can also be set up in the same menu – this time from the *Web Administration* tab. You must confirm each of these modifications by clicking on the *Save* button.

Check the time zone, date, and time settings on *System* | *Date & Time* and adjust these settings as necessary. Correct time and date settings ensure correct information in log files and are a prerequisite for some of the NAS automated actions. (See the "No Time" box.) The third step is to set up LAN access on *System* | *Network*. You can modify the configuration of the individual network adapters available on the system in the *Interfaces* tab and switch these services on and off in the *Service Discovery* tab. Finally, specify rules for the firewall as needed in the *Firewall* tab.

Moving to the last action for basic configuration of the OpenMediaVault system, you must install any pending updates. OpenMediaVault deploys Debian's Apt package system. Open *System* | *Update Manager* | *Updates*, displaying possible updates with *Check*, and install the updates that appear. Then activate the relevant options for automated updates in the *Settings* tab.

Plugins

OpenMediaVault, like most dedicated NAS operating systems, provides the option of integrating add-ons into the system. In the *System* | *Plugins* menu, you will find a wide range of additional applications sorted into sections. You can integrate any of these applications into the operating system with a mouse click.

Install an application by ticking the names and then clicking *Install* (from the list view). The system loads the desired program packages from the Internet and installs them. OpenMediaVault keeps the plugins at the same update level as the system by means of the update routine.

Disk Carousel

You must configure the existing mass storage for OpenMediaVault to function as a NAS system. Select *Storage* | *Physical disks* menu for a list of the devices connected to the system storage and their key technical parameters. It does not matter whether you are using hard disks or SSDs. However, the system does initially sideline mass storage devices that are connected via external USB interfaces. You must manually click on *Search* at the top of the list area to enable these devices.

The routine displays the disk drives in an overview table, along with technical specifications such as filesystem and memory usage. Drive-specific options such as the write cache or the power management settings can also be changed with a left-click on one of the drive letters, followed by *Edit* at the top of the list view.

You will find the *S.M.A.R.T.* menu item on the left under the *Physical Disks* entry (Figure 2). This item allows you to adjust the hardware monitoring for hard drives so you can anticipate and detect defects and secure the data saved on the drive in good time. However, please note that some external hard drives have their own controller electronics and will not fully implement the SATA interface commands. This phenomenon occurs especially frequently in external hard drives that offer further connectivity options in addition to the USB interface. Because these devices do not usually transfer any S.M.A.R.T. values to the host, checking routines do not run.

Activate the slider in the *Settings* tab of the S.M.A.R.T. menu, sliding it to the right and then clicking *Save*. Then select the drives to be monitored by means of the S.M.A.R.T. daemon in the *Drives* tab. The aim of this process is to manually activate monitoring for any mass storage devices connected; click on *Edit* and then confirm the activation rule. Your next move is to specify automated checking routines after clicking on *Add* in the *Scheduled tests* tab.

NO TIME

The Rasp Pi has never had an integrated clock (known in EDP jargon as a real-time clock, or RTC for short). The Pi therefore obtains the current time via an NTP server with each start-up. As a result of this limitation, OpenMediaVault displays a long error message on the Rasp Pi when applying changes in the settings. The developers recommend investing in an RTC module.

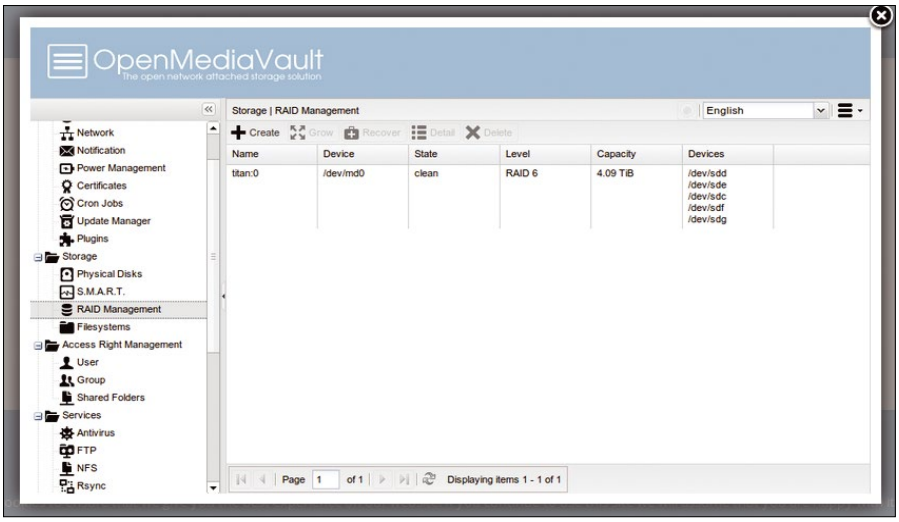


Figure 2: Choose configuration options in the tree view in the left pane of the OpenMediaVault main window. S.M.A.R.T. and RAID Management appear in the Storage folder.

It is worth creating a fault-tolerant disk array via the *RAID Management* menu item on a system with two or more hard disks. Click on *Create* in the display pane and define the RAID level, drives, and designation for the array in the resulting window. OpenMediaVault can allow the drives from a selection list by checking a box (with the exception of system drives). If there are too few physical mass storage devices for a particular RAID array, the software displays a note to this effect.

Click on *Save* after completing the RAID configuration. At this point, the system creates the disk array (as shown in Figure 2), which can consume quite some time, depending on the size of the mass storage device. You then generate a filesystem for the data carriers in the next menu item: *File Systems*. Select the array or the desired drive in the corresponding dialog and mark a suitable filesystem from the selection list. OpenMediaVault supports the Linux default filesystems Ext3/4, JFS, and XFS. You can close the dialog with *OK*; writing the filesystems takes some time.

Users and Groups

The next task is creating more users in OpenMediaVault by means of the *Access Rights Management* | *Users* dialog, which is accomplished by clicking on *Add* and defining your settings in the dialog box. After you have saved and applied the configuration, you can go on to create your own home directories for the newly-created users in the *Settings* tab. If you later wish to change the settings you defined, click *Edit* at the top above the application window's list view.

Users on OpenMediaVault belong to different groups. There are numerous out-of-the-box groups available on the system; you must select one or more of these groups when defining new users. If the existing group structure does not meet your needs and you want to add new groups to the system, use the *Access Rights Management* | *Group* dialog box.

Sharing

The final step is to define sharing and enable some services so that users working from desktop computers can access existing drives and directories in OpenMediaVault. Open the *Shared Folders* entry from the *Access Rights Management* menu and create a new shared folder by clicking on *Add* in the *Shared Folders* tab.

The configuration dialog allows you to specify a name and path. You can also assign permissions (for which the system clearly explains the various options). The button panel offers several other options for customizing the shared folder.

The most important of these options is the *ACL* button. A detailed dialog accessed lets you define specific access control lists (ACLs) for every single share, allowing fine-tuned allocation of rights (Figure 3).

After completing all the settings for sharing drives and folders, you can finish by setting up the desired background services (daemons) on *Services*. OpenMediaVault will only unlock access via SSH by default, so if you use the OpenMediaVault Rasp Pi in a heterogeneous network, you must enable Windows workstations access to the Rasp Pi through *SMB/CIFS*. A more complex dialog allows extremely detailed settings for SMB access.

You have the option of enabling other services, such as the ClamAV virus scanner. However, you must first install ClamAV via *System* | *Plugins*. The *Anti-Virus* entry will then appear in the left sidebar, which also branches out into a detailed settings dialog. OpenMediaVault allows actions from this box, including detailed scanning of various content and file types for malware.

Conclusions

The Debian-based OpenMediaVault and the RPi3 provide a neat NAS solution for small networks, which you can implement at a low cost. OpenMediaVault is excellently pre-configured and copes without any power-draining bells and whistles causing confusion for users. The clear and convenient user interface facilitates easy learning without wading through the technical literature.

However, don't expect high data rates from the Rasp Pi OpenMediaVault. If you really want speed, consider using an alternative single-board system such as the ODROID-XU4 [4], which features Gigabit Ethernet, two USB 3.0 ports, and eMMC flash storage. OpenMediaVault also offers ISO images for easy installation on the ODROID-XU4, as well as other ODROID models. ■■■

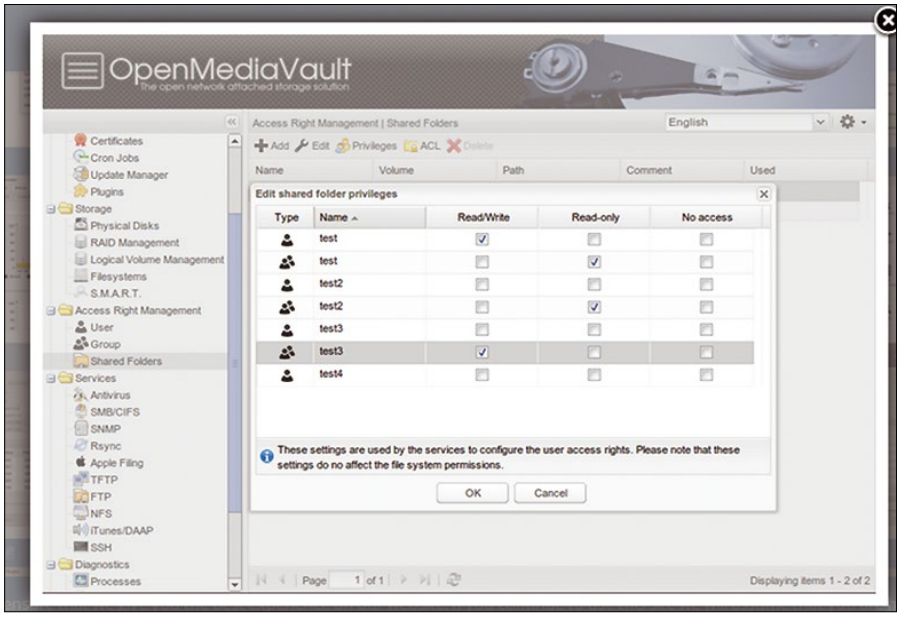


Figure 3: You can assign very detailed permissions with ACLs.

INFO

[1] OpenMediaVault: *http://www.openmediavault.org*

[2] Download: *https://sourceforge.net/projects/openmediavault/files/*

[3] Adafruit Raspberry Pi Finder: *https://github.com/adafruit/Adafruit-Pi-Finder*

[4] ODROID-XU4: *http://www.hardkernel.com/main/products/prdt_info.php?g_code=G143452239825*

Optimal DIY NAS with Rockstor Linux

Tailor Made

Rockstor Linux turns a microserver into a fully functional NAS. *By Erik Bärwaldt*

To help cope with fast-growing volumes of data, more and more users are installing network-attached storage (NAS) systems on their own networks. Several hardware vendors would love to sell you a dedicated NAS device, but if you don't have the budget for expensive, proprietary NAS hardware, you can still get in the game. A compact PC with state-of-the-art hardware is ideally suited for network data storage, assuming it meets a handful of conditions. I will show you how to configure a home-built NAS system with a compact computer running Linux.

Requirements

For your custom NAS, you don't need a fast processor, and you can even use an older system that is no longer suitable as a desktop system. However, other considerations, like the mass storage subsystem and the performance of the power supply play an important role in NAS.

Thus, you should make an investment in your DIY project: Old computer systems that still rely on IDE interfaces using the parallel ATA standard only support two hard disk drives or SSDs per connection. Years ago, the parallel ATA interface was replaced because of poor performance with the more modern serial ATA (SATA) standard.

The SATA specification not only provides for far higher data transfer rates, it also allows far larger media; therefore, the cumbersome, error-prone master/slave configuration does not apply. If you do intend to use an older computer system for building your NAS, make sure it at least provides the SATA 2.0 (3Gbps) standard interfaces.

One problem in building a NAS system from existing components is rooted in the power supply. Legacy models usually have a relatively high power consumption and are designed for the CPUs of that time, which were not exactly economical. Because a NAS is usually intended for continuous operation, you can expect significantly higher costs if you use such components. Therefore, it is not advisable to use power systems designed for Pentium 4 processors or the first generation Dual-Core-branded processors. Newer power supplies, designed for 80 Plus certification, will ensure greatly increased efficiency.

Finally you should look for hardware that lets you access storage slots easily from the outside. Commercial systems have replaceable bays with corresponding mechanisms, so a defective disk can be replaced within a few minutes. Conventional cases generally do not have slots that let you exchange components, but you might be able to add them retroactively.

When using more than two hard drives, the use of a RAID controller offers better data integrity in the case of hard disk failure – if you select an appropriate RAID level. A RAID controller can be quite expensive, so you need to calculate precisely before the project to see if it is actually still worthwhile upgrading existing hardware for the desired purpose or if purchasing a compact system will be cheaper.

Because the configuration of software RAID without dedicated controllers can now be managed easily, especially with regard to the mass storage devices, precise

planning of the kind of data security you want to achieve is advisable. The simple software solution might already meet your needs.

ProLiant MicroServer

The Hewlett Packard Enterprise (HPE) ProLiant MicroServer Gen8 [1] is optimally designed for use as a NAS and comes standard with all the necessary components in place from the factory. You can purchase the device in various configurations, with processors from a Celeron running at 2.3GHz clock frequency up to an identically clocked Xeon.

A smart array controller with four bays for hard drives or SSDs is common. The configuration lets you run various RAID levels with hardware support. The bays with the drive cages are fitted vertically at the front of the unit behind a cover, but are not designed for hot-swapping, although they can hold at least 16TB of mass storage. You need an adapter if you want to install 2.5-inch drives. The first two slots use the current SATA 3.0 standard with maximum speeds of 6Gbps, whereas the other two only support the older SATA 2.0 specification.

The device also has four external USB ports that support the current USB 3.0 standard and three RJ45 LAN connectors, one of which is designed as an integrated lights-out (iLO) connector, which, in case of a problem, allows remote management of the server. The LAN interfaces all support Gigabit Ethernet.

Inside the server on the motherboard is a slot for SD/SDHC memory cards and a USB port. The internal connections are bootable, which means you can use media inserted there to boot operating systems. (See the "Bootable USB Sticks" box for more information.)

Another slot for an optical drive with a slim form factor is accessible from the outside. On the main circuit board of the computer are two slots for standard DDR3 memory modules and one free PCIe slot for additional cards, which you can access without tools. The graphics card is a Matrox G200, which sends signals via VGA output with resolutions up to 1920x1200 pixels and a color depth of 16 bits/pixel.

Rockstor Linux

As the operating system for the home-built NAS, I will use Rockstor Linux, which the developers optimized for use as a NAS. You can get the free system – based on CentOS and Fedora – from the project page in the form of an approximately 730MB ISO image [2].

Compared with a dedicated NAS system based Linux, Rockstor comes with Btrfs as the default filesystem and implements many advantages of the ZFS filesystem [3] already used by some distributions. The system is also suitable for use in heterogeneous IT environments because it supports SMB/CIFS.

The distribution also uses Docker containers, called "Rockons" in this context, that let you integrate cloud services into the system. You can configure and manage the system in a customized JavaScript interface, which means you can access it at any time from any computer on the LAN.

Preparation

The ProLiant server initially has an unusually long initialization phase. However, this ensures that all components are doing what they are supposed to be doing. To address the mass storage devices installed in the computer properly using Rockstor, you need to configure the SATA controller built in to the server before installing the operating system.

Pressing F9 during the initialization phase runs the graphical setup, in which you select the first entry, *System Options*, by pressing the Enter key. Next, select *SATA Controller Options*, then open the *Embedded SATA Configuration* dialog by pressing the Enter key.

In the new window, select the option *Enable SATA AHCI Support*. This turns off the B120i RAID controller, and the mass storage is now under the control of the Intel SATA controller. Now

BOOTABLE USB STICKS

More and more motherboards have internal USB ports; SD/microSD slots can occasionally be found on the motherboard, as well. These ports are marked in the BIOS as bootable, so you can connect a boot drive in the form of a USB stick or a memory card. Rockstor Linux also supports USB media as boot drives, but it is advisable not to use conventional USB flash drives. Many writes to the devices take place, especially with system logging, and because the memory cells of commercially available USB sticks support only a limited number of write cycles, you should choose a USB flash drive for the boot drive that relies on single-level cell (SLC) technology. Compared with the usual multi-level cell (MLC) or even triple-level cell (TLC) flash memory, SLC can handle many more write cycles, it is significantly faster, and it has a service life similar to SSDs.

quit the BIOS by pressing Esc several times, until all the dialogs are closed. To save the new settings before rebooting, press F10.

This procedure is required to install Rockstor onto mass storage. Because controlling the RAID controller requires a proprietary driver from Hewlett Packard, which is missing from the standard Rockstor, the system would not cooperate with the controller. The driver available from Hewlett Packard for the B120i RAID controller in the network is certified for SUSE Enterprise Server and Red Hat, but that doesn't mean it automatically supports all CentOS- or Fedora-based systems: Although Rockstor let me install the HP driver, it did not work properly.

After customizing the BIOS and restarting, you end up in a simple GRUB boot manager, which has an option for installing the operating system directly or after testing the medium. The third option, *Troubleshooting*, takes you to a submenu where you can repair a damaged system or install the system with a general-purpose graphics driver.

Because the NAS system will not have partitions – just complete mass storage devices – you should install the operating system on a USB stick if you are only using two hard disk drives or SSDs, so you can add both internal media to the storage pool.

In most cases, the default installation should work fine: Server systems rarely use extremely exotic video cards, which largely prevents problems with the modules. After selecting one of the options, the Anaconda graphical installer launches.

In the first dialog, you select the locale and set the time zone. Next, you define the disks on which to install the system; the routine will already have selected some sensible defaults. For the root filesystem, Rockstor Linux exclusively supports Btrfs; you also set the parameters for the network interface in this dialog.

The routine detects only wired access. In the case of the server, it is possible to address one of the two Ethernet ports. The software ignores WiFi hardware; as a result, the dialog offers no way to configure it. During the subsequent installation, you create a user account for the operating system administrator. The installer copies the system to the mass storage device in parallel.

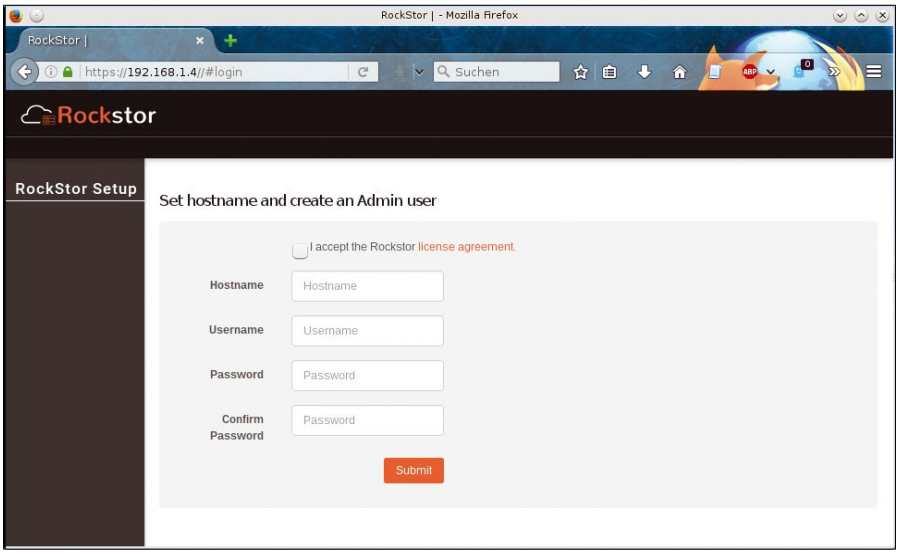


Figure 1: Rockstor Linux initially expects you to create an account for the administrator.

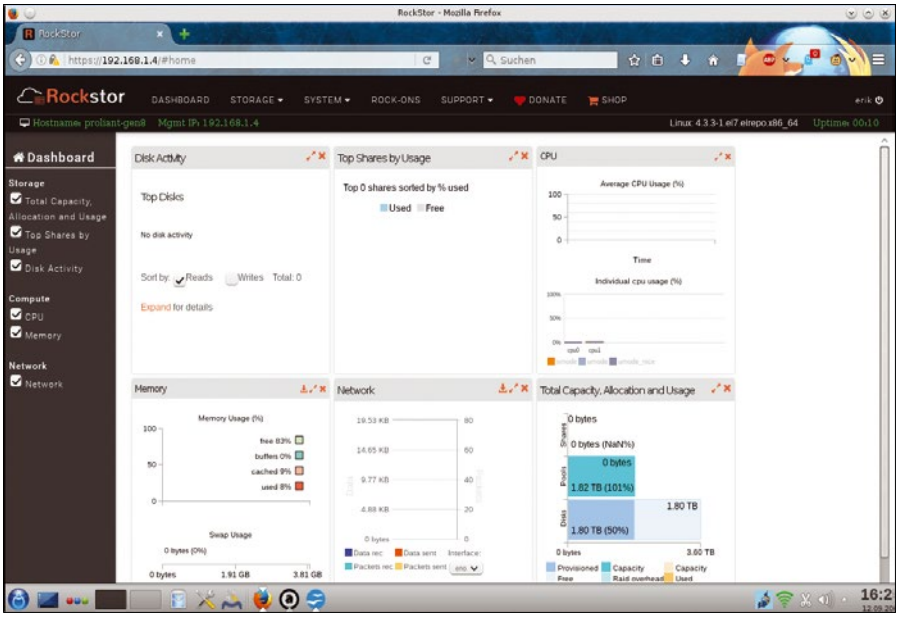


Figure 2: The dashboard lists all the important activities and the status of the system.

After completing the installation and rebooting, the system tells you before you log in that you can access the graphical configuration of the server, if required, from a web browser on any computer on your intranet by typing the URL *https://<IP address>*. Calling the address in a browser branches to a simply designed page (Figure 1). Here, you first create the Hostname of the server, as well as an account for the administrator of the management interface.

Now, you are taken to a dashboard, which provides a wealth of information summarized in groups (Figure 2). Among other things, it displays the status of the CPU, memory, mass storage, and network interface, even showing the throughput. A horizontal menubar at the top gives you access to more settings.

Getting Started

To use Rockstor Linux, you first need to create a pool and shares. The menu item *Storage* | *Pools* | *Create Pool* in the dashboard takes you to a dialog where you can define new pools, for which you also set the appropriate RAID level (Figure 3). If only one mass storage device is available, just set the Raid configuration box to *Single*. If you have several disks, choose a RAID level of 0, 1, 10, 5, or 6.

Clicking *Submit* completes the process, and the system displays a corresponding entry in the table. Then you can use the *Storage* | *Shares* | *Create Share* dialog to set up shared directories, while defining compression for the respective shares. After clicking *Submit*, the shares are available.

To use the shares, you need to publish them on your intranet. You have several options for doing this depending on the client operating systems: In addition to NFS shares, it makes sense to enable Samba shares for Windows machines. The Rockstor system is also capable of supporting SFTP or AFP for Apple environments.

All of these options can be reached in the *Storage* | *File Sharing* menu. First, you need to enable some of the services, such as NFS and SMB, in the corresponding dialogs by flipping a virtual switch; without this feature, you would have to install the software.

The individual network filesystems' options also let you define the permissions and specify which hosts or subnets have access. You can then start using the NAS system as a drive in the file manager of your workstation (Figure 4).

The *System* | *Identity* menu, with the subgroups *Users* and *Groups*, have dialogs structured similarly to the *Storage* option, giving you granular control of access privileges. Here, you create and manage your user accounts and groups.

Additions

Much like the dedicated NAS systems with proprietary operating systems, Rockstor Linux lets you integrate additional modules, which come as Docker containers. To install these modules, go to the *System* | *Services* menu and enable the *Rock-on* service by clicking the slider to *ON* (Figure 5). The system prompts you to release a share specifically for the additional components and then enables the service. On the dashboard, you will now see the available applications in the *Rock-ons* menu. Although the selection is not as extensive as with a dedicated NAS, it extends from file-sharing services, to the free ownCloud solution, to media servers and databases.

Installing the desired application is just a matter of pressing the *Install* button; the system creates a basic configuration before installing, depending on the service, that integrates the respective daemon (Figure 6). In the same menu, you can turn off existing applications and, if necessary, uninstall them. To do so, go to the *Installed* tab and push the slider to *OFF*; then, click on the *Uninstall* button.

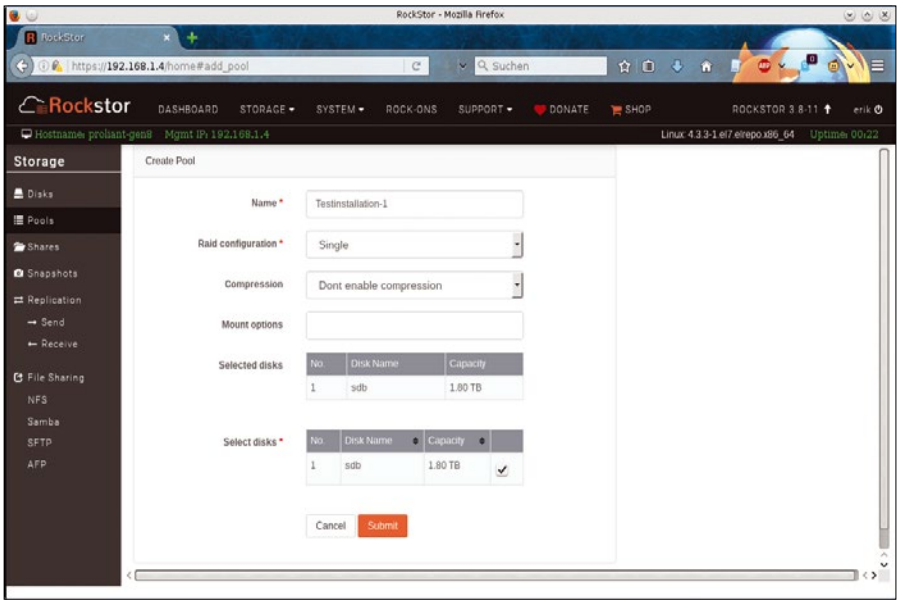


Figure 3: With just a few clicks of the mouse, you can create a pool quickly via the web interface of the specialist distribution for NAS systems.

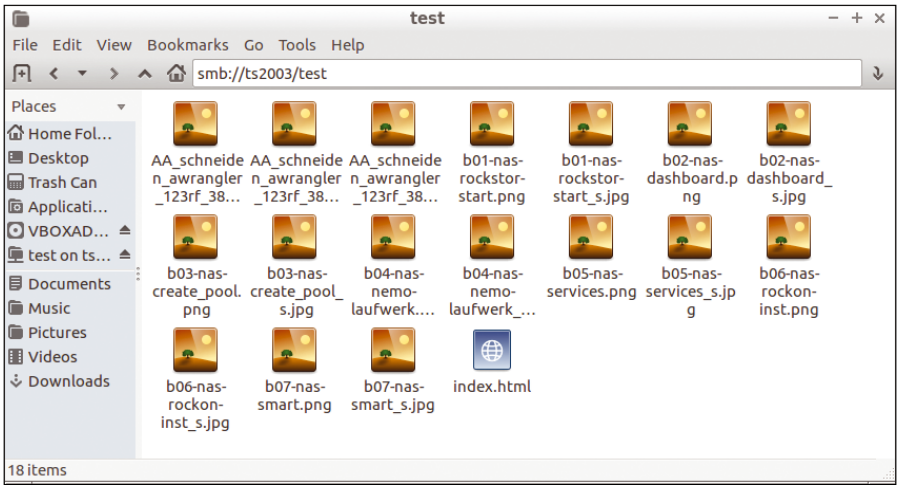


Figure 4: Using the desktop file manager, you can mount the NAS on the client.

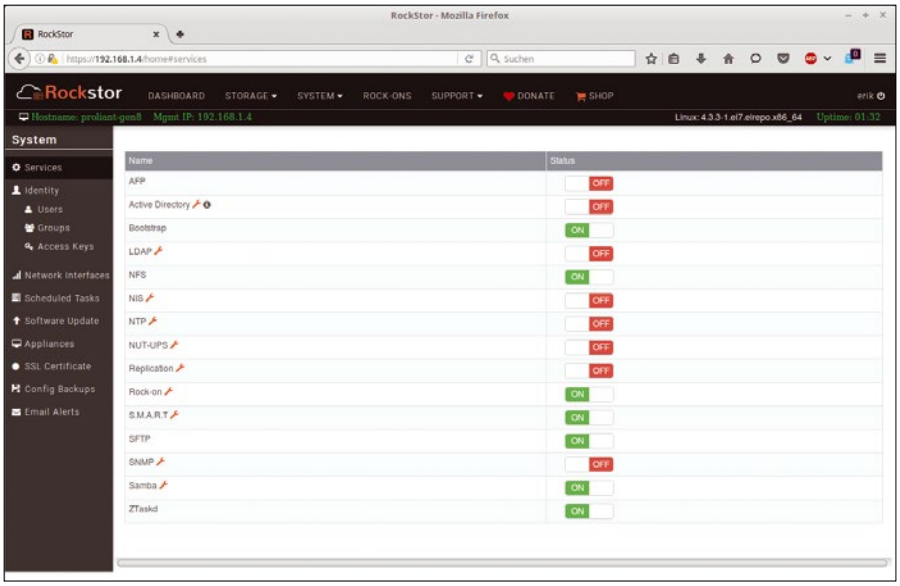


Figure 5: An additional service manages the extensions that reside in Docker containers and for which you need a dedicated share.

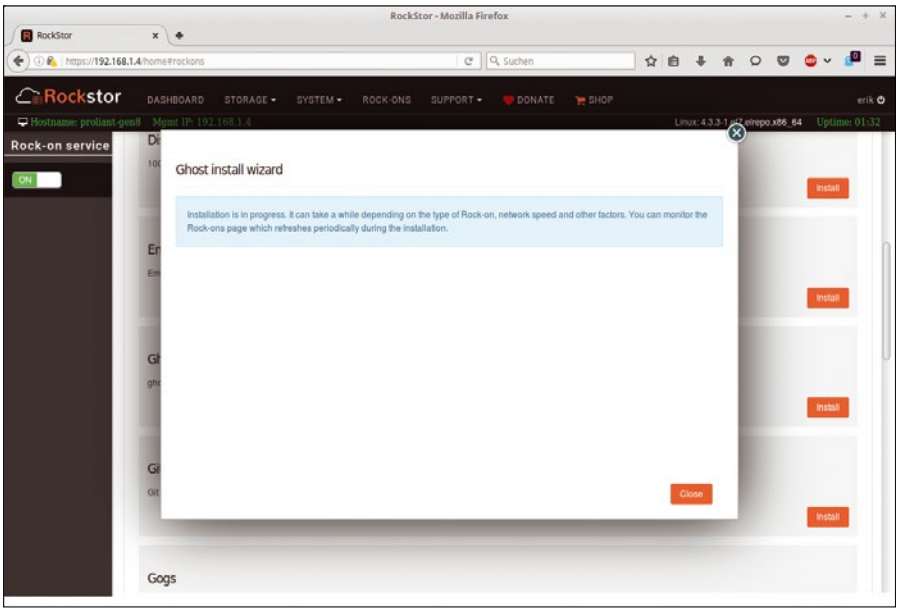


Figure 6: Rock-ons are integrated into your system automatically with a few clicks of the mouse.

Data Security

Rockstor provides security against data loss through different components. For example, you can create snapshots of the installation in the *Storage* | *Snapshot* menu and back up your data with automated backups. The *System* | *Scheduled Tasks* menu also lets you define fixed periods for when backups expire.

The developers have also thought of the hardware. By default, the operating system switches on the S.M.A.R.T. service – if it is supported by the hardware – to monitor your storage, which preemptively finds hardware defects. If S.M.A.R.T. is enabled, you have the option of going to the *Storage* | *Disks* menu to retrieve data or perform tests by clicking on the desired drive and then selecting one of the tabs. A click on the *Refresh* button at top right executes the desired function in the window (Figure 7).

Conclusions

Rockstor proves to be both a sophisticated and a flexible NAS system for home-built servers, thanks to the use of Docker containers and Btrfs. The simple operating concept and the uncluttered interface go hand in hand with good stability. Thus, Rockstor is highly recommended for use in everyday life. For users who want to rehash no longer used legacy hardware as a central storage system or convert a microserver into a NAS, the free NAS operating system provides an interesting solution. ■■■

INFO

[1] HPE ProLiant MicroServer Gen8: *http://www8.hp.com/us/en/products/proliant-servers/product-detail.html?oid=5379860*

[2] Rockstor: *http://rockstor.com/download.html*

[3] Btrfs: *https://btrfs.wiki.kernel.org/index.php/Main_Page*

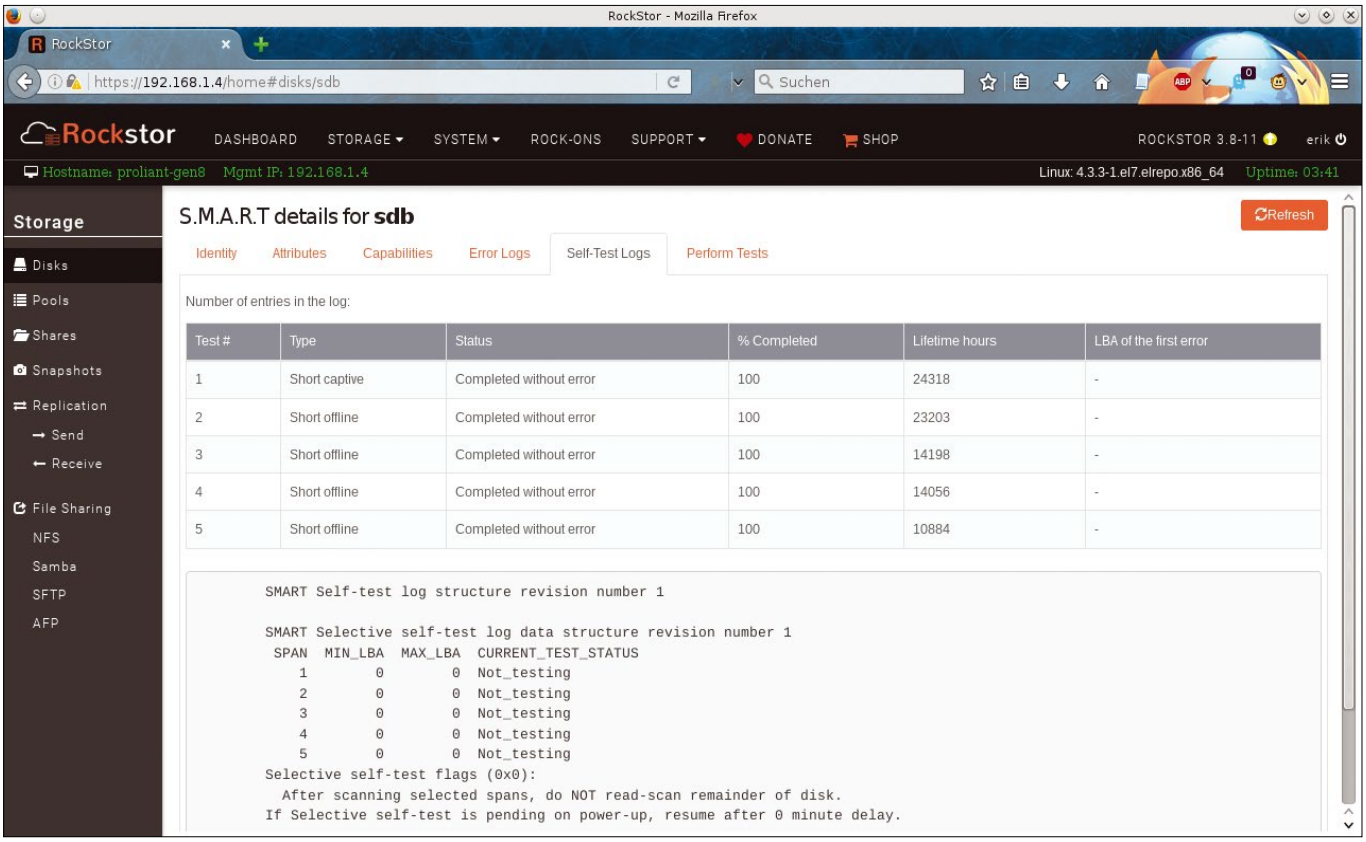


Figure 7: Rockstor Linux keeps an eye on your hardware.

SRECON®

SREcon is a gathering of engineers who care deeply about site reliability, systems engineering, and working with complex distributed systems at scale. It strives to challenge both those new to the profession as well as those who have been involved in it for decades. The conference has a culture of critical thought, deep technical insights, continuous improvement, and innovation.

AMERICAS
MAR 13–14, 2017
SAN FRANCISCO, CA
www.usenix.org/srecon17americas

ASIA AUSTRALIA
MAY 22–24, 2017
SINGAPORE
www.usenix.org/srecon17asia

EUROPE MIDDLE EAST AFRICA
AUG 30–SEP 1, 2017
DUBLIN, IRELAND
www.usenix.org/srecon17europe

Linux Lite 3.0 for legacy systems

Travel Lite

Lean Linux distributions compete for the favor of users with older hardware. Linux Lite enters the scene with some interesting options for achieving simplicity with full functionality. *By Erik Bärwaldt*

Older computers with slow processors and minimal RAM don't work so well with mainstream desktops and bulky graphical applications. Several Linux distributions are designed to appeal to the users of older systems, offering practical yet slimmed-down apps and kernels tuned for a more minimal setting.

New Zealand's Linux Lite [1] is a lean Ubuntu derivative aimed at Linux newcomers. In addition to a lean design and conscious avoidance of bloat, Linux Lite offers some simple yet effective management tools that will appeal to transitioning Windows users and other Linux beginners.

Since Linux Lite 3.0 is based on the current Ubuntu 16.04 version with LTS support, it will receive security updates for several years.

The hardware requirements are very modest: Linux Lite is content with a 700-MHz CPU and 512MB RAM; the monitor should be able to display at least 1024x768 pixels. Even older computers with 4:3 displays, such as elderly notebooks, can thus easily cope with Linux Lite.

The 950MB 32- and 64-bit Linux Lite images are available from the project website [2]. You can use these hybrid ISO images to create a bootable DVD or USB stick.

Getting Started

Burn the bootable image to a DVD or USB stick. After you boot the Linux Lite ISO, you can either test Linux Lite in Live mode or launch a permanent installation from the Grub menu. In Live mode, Lite starts up surprisingly quickly, showing a fully operational XFCE desktop. In fact, you have to look closely to realize the visually appealing system is actually the simple and light XFCE. At first glance, you would swear it is KDE.

Figure 1: The Welcome screen introduces the system.

Right after booting, Linux Lite displays a welcome screen and introduces the system (Figure 1). Clicking on one of the buttons reveals information about how the operating system works and what options it offers.

The Linux Lite developers designed the *Help Manual* as a very practical aid: Clicking on this button launches the Firefox web browser and opens the locally installed manual, which provides information and assistance for the many configuration options. The instructions also contain illustrations of the current menus, so that you will quickly find your way around (Figure 2).

Only three icons appear on the Linux Lite desktop in Live mode: In addition to a symbol for the built-in mass storage devices, you'll find an icon to let you call the manual, as well as the *Install Linux Lite* icon, which calls the Ubiquity installer familiar to Ubuntu users. Ubiquity installs the system on your hard drive in just a few steps.

The XFCE desktop looks very tidy with its classical menu. Strong color contrasts and large, intuitive symbols allow users to keep working without delays, even under adverse lighting conditions, such as notebook displays with high ambient light.

Fortunately, Linux Lite does without annoying, resource-consuming gimmicks

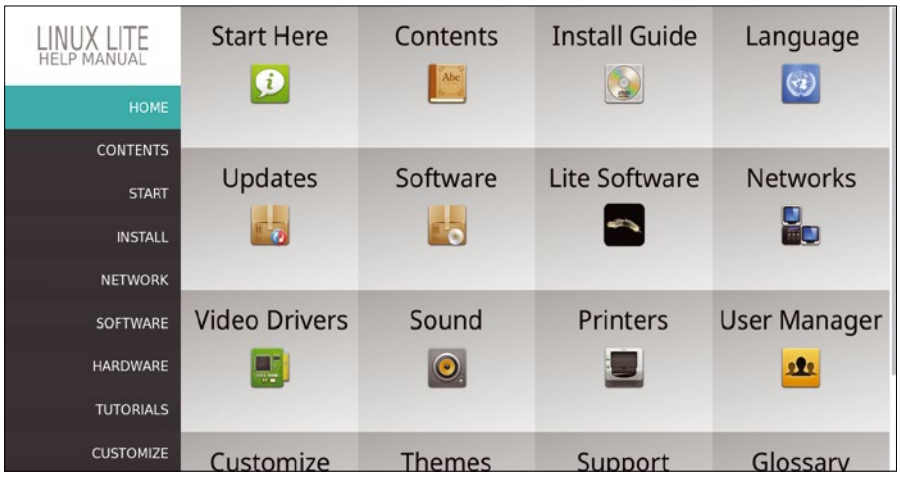


Figure 2: The Help Manual familiarizes users with the system.

such as wobbling or crumbling windows. You will not even find a shadow around the edges of the window or other subtle effects, such as intrusive acoustic feedback, which tends to disrupt the workflow.

Software

A quick look at the main menu and its branches shows that Linux Lite comes with a full set of important applications, in spite of the ISO image's relatively small size. On board, you will find GIMP 2.8.16, LibreOffice 5.1.2.2, Firefox 46.0.1, and Thunderbird 38.8.0, as well as the all-round Media Player VLC 2.2.2. Also included are some smaller applications from the XFCE treasure trove.

Rounding out the application set is a useful collection of system programs: For example, the *Settings* | *Firewall configuration* submenu contains the Gufw tool, which provides a convenient graphical configuration option for the firewall. The *System* | *Systemback* entry offers an easy backup tool. In addition, the *Accessories* menu has *Backups*, another tool for enabling automated data backups on a customizable schedule.

In-House Development

Linux Lite is not confined to the role of a visually enhanced and resource-friendly Ubuntu clone: The system also comes with its own applications, which complement the existing functionality and contribute to the general ease of use. These tools include Lite Tweaks, Lite Software, Lite User Manager, and Lite Upgrade.

Linux Lite does not have its own software store but relies on the Synaptic GUI and installs from the Ubuntu repositories. The extensive Ubuntu repository system provides more than 50,000 packages, which are ready to install on your mass storage media. Lite Software, which you will find in the *System* menu, complements the software manager with an interesting alternative: It lets you install the most frequently used packages at the push of a button, without long searches; the tool automatically resolves all dependencies (Figure 3).

Newcomers, who often find Linux confusing, will appreciate the Lite Software option as a simpler way to manage packages. Lite Software does not implement a new package management system but uses existing resources. The most popular packages that you can install with Lite Software include Audacity, Calibre, Chromium, Kodi, PlayOnLinux, TeamViewer, the Tor web browser, VirtualBox, and Wine.

Lite Upgrade performs a complete upgrade of the operating system after a new Lite release. You can launch the preinstalled Lite Upgrade tool with the `lite-upgrade-series3` command to discover when the next system refresh is intended (Figure 4).

The tweak tool Lite Tweaks (Figure 5) automatically optimizes the system. You will find the graphically simple Lite Tweaks in the *System* menu. Lite Tweaks mainly handles the task of cleaning up obsolete files on the hard drive. You can also clear the current Firefox cache or check the options to initiate operating system-specific actions, such as deleting the package buffer.

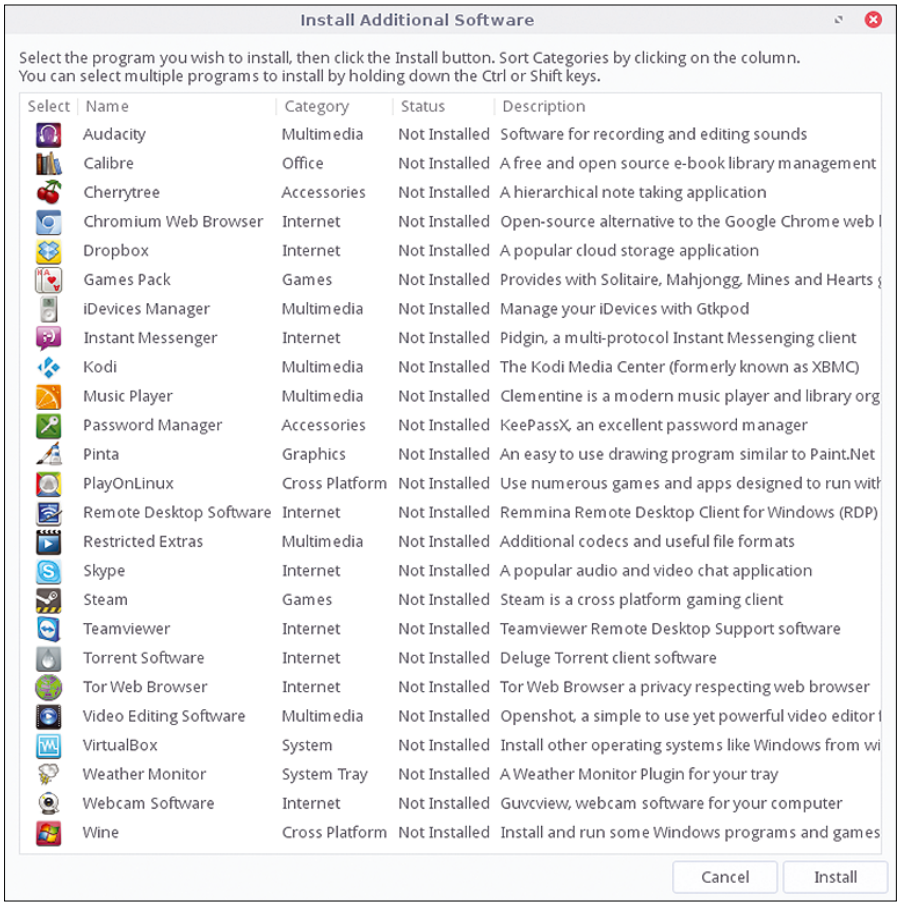


Figure 3: The Lite Software tool installs the most frequently used applications on your local mass storage device with just a few mouse clicks.

To prevent unconsidered deletions that could lead to malfunctions or even make the system inoperable, the developers added a *Status* column to the display. The *Caution* entry shows that caution is advisable before deleting the selection. The *Task* column shows the effect the modification will have on the system.

A *Clean* entry in the *Task* column cleans up the system by deleting unnecessary files, and *information* points to useful information about mass storage devices or very large files. In the *Description* column, Lite Tweaks shows how much disk space you would save by triggering the delete action.

Figure 4: Now you know exactly when the next system upgrade is scheduled.

A Management Thing

Linux Lite offers various tools for convenient graphical management of the system. It cleanly separates the configuration of the actual operating system and its components from the components of the XFCE desktop. For example, you will find tools for software management, printer configuration, partitioning, and task management.

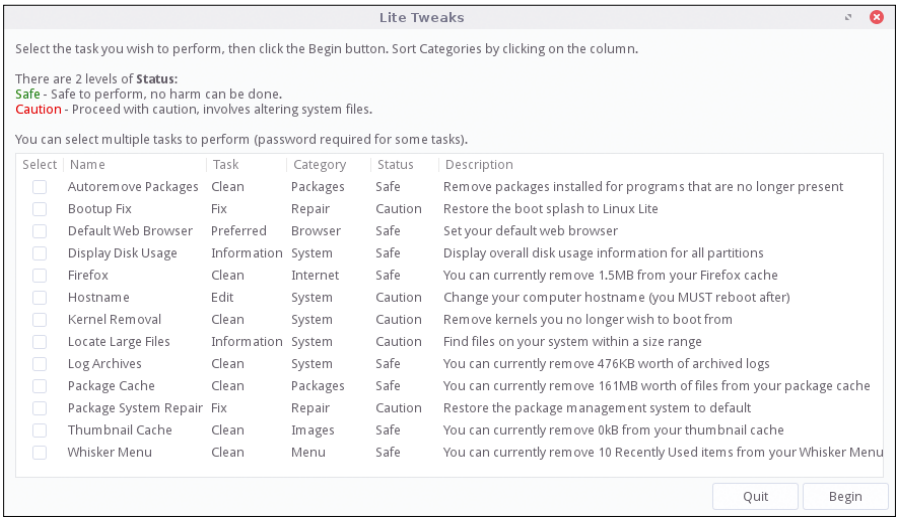


Figure 5: Lite Tweaks helps you keep your system trim and lean.

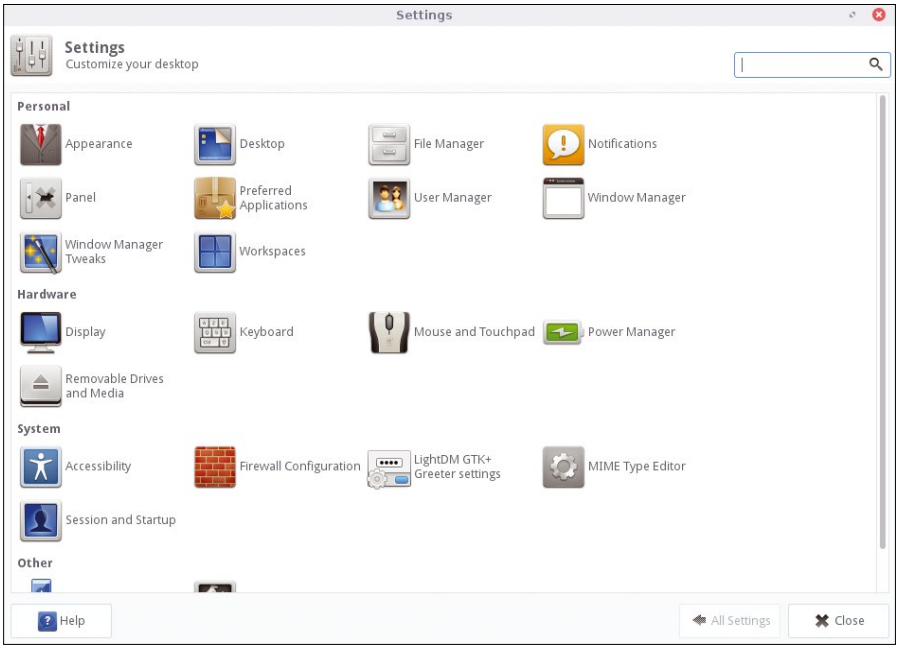


Figure 6: The XFCE Settings Manager provides tools for managing the XFCE desktop.

The *Settings* menu includes the tools for the desktop configuration. This includes, in particular, the integrated XFCE management interface *Settings*, which primarily defines the appearance of the desktop environment and specifies how removable drives, input devices, and services are handled (Figure 6).

The individual options grouped in the *Settings* tool can be accessed via the *Settings* menu, so you do not need to go through the tool in many cases. The XFCE desktop offers a perfect balance in terms of configuration options: You cannot drill down to the smallest detail, as you can with KDE, but you can access significantly more options for individual modifications than in newer versions of Gnome.

The Linux Lite *Settings* menu includes tools for the network configuration, including the ability to customize the firewall graphically without in-depth knowledge of iptables. *Install drivers* is a graphical tool for addressing hardware with proprietary drivers. The *Install drivers* tool proves especially useful with recalcitrant WLAN and UMTS cards, and if you need to support laptop graphics chips: The installation of proprietary drivers and firmware files can easily be automated without the need to search the vendor pages and manually load OEM modules.

Conclusions

Linux Lite has an impressively efficient approach and excellent hardware support on older machines. For example, Lite ran very smoothly on laptops that were between five and eight years old, some of which had hardware that is hard to address with Linux.

And thanks to Ubuntu's extensive software repositories, you can customize Linux Lite for virtually any use case. The lean, but by no means boring, XFCE desktop ensures an appealing look and supports efficient work, thanks to good ergonomics. Users migrating from other operating systems, and newcomers who want to avoid a long learning curve, are well served with Linux Lite. ■■■

INFO

[1] Linux Lite: *https://www.linuxliteos.com/*
[2] Download Linux Lite: *https://www.linuxliteos.com/download.php#current*

Plasma 5 with long-term support

Plasma Zone

KDE is steering the Plasma 5.8 desktop into calmer waters through long-term support, while the developers continue to extend and perfect. *By Ferdinand Thommes*

When Matthias Ettrich created the *Kool Desktop Environment* more than two decades ago [1], no one knew where the journey was heading. Ettrich had noticed that the already five-year-old Linux was becoming increasingly sophisticated, but it lacked a uniform and stylish desktop environment that made daily work easier for end users.

For more than 20 years and five generations of software, KDE has remained true to its task of supporting end users, whereas Gnome, which is three years younger, increasingly targets developers and other professional users. Ettrich's ongoing commitment led to some personal recognition: He was awarded the Medal of Merit of the Order of Merit of the Federal Republic of Germany in 2009 for founding the KDE project.

Tripartite

KDE comprises three parts: KDE Desktop; KDE Frameworks, which was previously monolithic KDE libraries; and KDE Applications. The desktop component has gone by the name Plasma since the fifth incarnation. Although the three components interact, they do not need to be released together. This independence makes work easier for developers.

Plasma is currently at version 5.8 and, as the first LTS version in the fifth cycle, now offers long-term support with security updates and bugfixes for at least 18 months [2]. The same is true for KDE Frameworks 5.26. The extended support will make it easier for enterprise Linux distros to offer continuous support for the lifetime of the release.

Plasma 5.8 LTS was released in early October; most of the remaining errors from version were 5.7 eliminated, and some changes were introduced.

Task Manager

The previous Plasma 5.7 introduced a task manager back end [3]. The previous code was already 16 years old and designed for X11, which made it difficult to use with the new Wayland display manager. In Plasma 5.8, some applets that use the new back end have been completely revised; in particular, the pager and the window list have seen extensive changes. They now obtain their data directly from the task manager, which saves memory and conserves resources.

Even the task manager itself gets some new features with Plasma 5.8. You can now drag files onto a linked application in the task manager and open them along with the program. The context

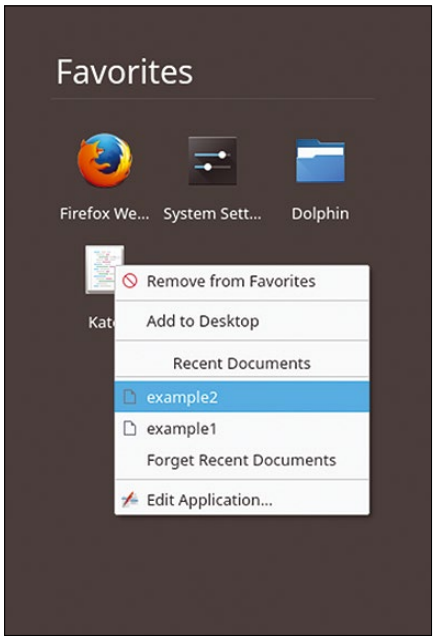


Figure 1: The Kate jump list in the alternative Dash menu.

menu of applications in the task manager now also supports media control. With a right-click on the application in the bar, you can control the volume of music and movies or skip to the next track in a playlist.

Jump Lists

One major simplification with Plasma 5.8 involves how global keyboard shortcuts are handled [4]. Actions can now be called using a single modifier key like Ctrl, Alt, or Shift without having to press another key. The Windows button, also referred to as the Meta key on Linux, now defaults to opening the K menu as a dash in full-screen mode. This innovation is currently not reflected in the *Shortcuts* part of the system settings; you need to configure it manually in `~/.config/kwinrc`. Jump lists introduced in Plasma 5.6 make it possible to open applications with a direct jump to specific functions (Figure 1). They now also allow jumps within a single application at the push of a modifier button, which means you can open, say, the Compose window of an email client or a web browser in private browsing mode [5].

Multimonitor

Plasma 5.8 finally puts an end to the problems that existed in earlier Plasma versions with controlling multiple monitors. The same applies to the use of projectors or docking stations. On the software side, Plasma now uses Spectacle instead of KSnapshot for creating screenshots (Figure 2). Plasma Discover, a tool for targeted searches for applications, has been newly developed in line with the specifications of the Kirigami framework (Figure 3).

A few improvements to the Save dialog in Spectacle would be helpful: The dialog does not include the *Places* display in Dolphin and requires several mouse clicks more to access frequently used storage locations.

Major upheavals unnoticed by users have been taking place under the Plasma hood for some time now. These changes mainly concern the KWin window manager, which the developers have been preparing for the new Wayland display manager [6]. For example, the KDE window manager under Wayland already supports all the common features found in X11.

Future

The KDE developers are still working on bringing the full KDE4 functionality to Plasma 5.X, but a quick look at KDE developer Sebastian Kügler's plans [7] reveals that, for example, the global menus are set to return in a contemporary form as early as January 2017 with Plasma 5.9. This change will mean that application menus will appear outside of the application window, as in Ubuntu's Unity: at the top of the screen or in the system tray.

Additionally, just three Plasma releases per year are planned for the next two years. The developers hope for a better balance between development work and stabilization phases between the releases. Plasma 5.9 is set for release January 31, 2017, and the next LTS version is scheduled for August 2018 in Plasma 5.14. The next LTS version will correspond with another LTS version of the underlying Qt framework, which is planned for the summer of 2018.

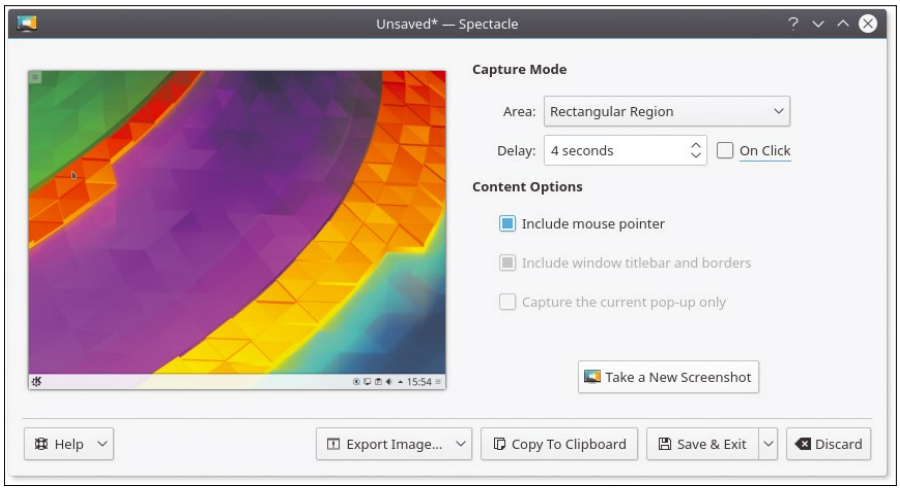


Figure 2: Spectacle has replaced KSnapshot as the screenshot application.

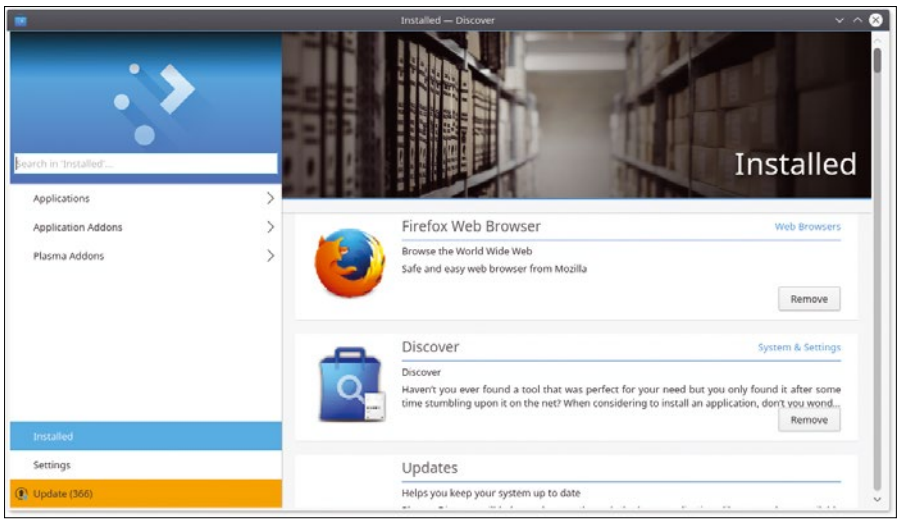


Figure 3: Muon Discover provides structured information about available applications.

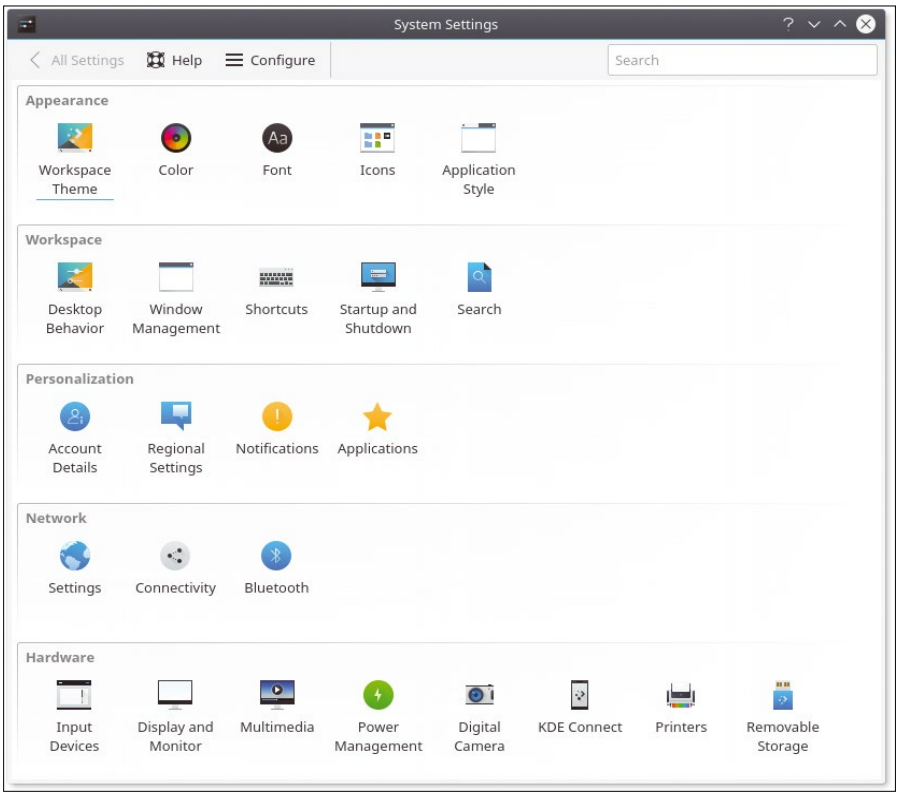


Figure 4: The tidy System Settings window provides easy access to configuration options.

Problems

KDE still faces one problem: The PIM suite Kontact and KMail have fallen even further behind the general development and are still fraught with many errors. Back in KDE4 days, two new technologies were introduced to KDE that caused many problems for Kontact and KMail: the Nepomuk semantic desktop and the Akonadi datastore, which provide PIM applications such as email and calendar programs access to uniformly managed data and metadata. Nepomuk has now been tamed by the new index and search framework Baloo, but Akonadi is still having problems, displaying complexity that does not fit well with the ambitions of an intuitive PIM suite.

Gnome and KDE have shifted increasingly away from each other in recent years, with Gnome becoming more simplified, leaving out tried and trusted features of the past. The fifth generation of KDE continues to polish its claim of being configurable throughout, but usable with the defaults. KDE has sacrificed nothing of this vaunted configurability in Plasma 5. On the other hand, the default configuration meets most requirements, without confusing or missing too much.

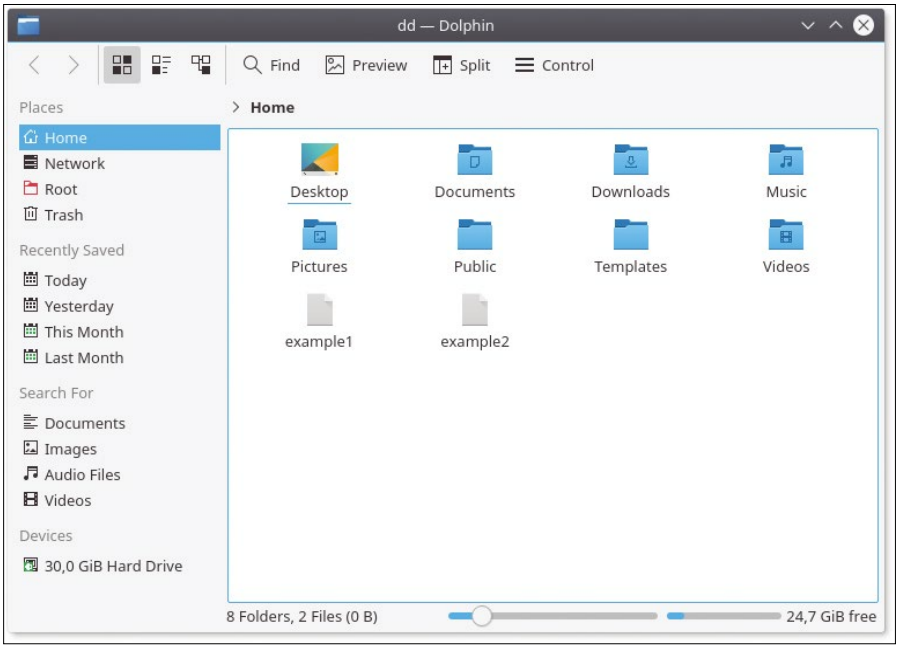


Figure 5: Dolphin from Plasma 5.8 in openSUSE Leap 42.2 RC1.

Conclusions

The latest KDE is the best that ever existed in the eyes of many users. You might not like the default wallpaper, but you can replace it in seconds if you are not satisfied. All told, the visual appearance is clear, elegant, and businesslike at the same time; the design retires to the background and is unobtrusive (Figure 4). With Dolphin as its control center, KDE has undoubtedly one of the best file managers on the market (Figure 5).

With Plasma 5.8 LTS, KDE finally seems to have settled into the fifth cycle. The KDE developers have already treated Plasma 5.8 to two minor updates, which are available in cutting-edge distributions such as KDE Neon [8] or the small but impressive KaOS [9]. With Plasma 5.8, KDE now seems more than ever to know where the project is going. ■■■

INFO

[1] Announcement: *https://groups.google.com/forum/#!original/de.comp.os.linux.misc/SDbiV3lat_s/zv_D_2ctS8sJ*

[2] Plasma 5.8 LTS: *https://vizzion.org/blog/2016/09/lts-releases-align-neatly-for-plasma-5-8/*

[3] Task manager: *https://blogs.kde.org/2016/05/31/new-plasma-task-manager-backend-faster-better-wayland*

[4] Keyboard shortcuts: *https://blog.martin-graesslin.com/blog/2016/08/modifier-only-shortcuts-available-in-plasma-5-8/*

[5] Discussion on jump lists: *https://forum.kde.org/viewtopic.php?f=285&t=129536*

[6] Wayland in Plasma: *https://blog.martin-graesslin.com/blog/2016/06/wayland-in-plasma-5-7/*

[7] Plasma roadmap: *https://vizzion.org/blog/2016/10/plasmas-road-ahead/*

[8] KDE Neon: *https://neon.kde.org/*

[9] KaOS: *https://kaosx.us/*

Encrypt files and folders with TruPax 9

Wrapped

The TruPax tool specializes in encrypting small datasets to safeguard your data from prying eyes. *By Erik Bärwaldt*

Cryptographic software tools are two a penny on Linux, but complete encryption of a disk or partition is hardly worthwhile for individual small datasets. On the one hand, it takes a long time to complete the operation depending on the physical size of the storage medium; on the other hand, encrypted partitions are no longer portable. One remedy is cryptographic software that bundles the data into volumes of variable sizes. In this article, I look at TruPax 9, a simple but useful application for home use.

The standard solution – the undisputed King of the Hill, at least on Linux – was TrueCrypt until the developers surprisingly stopped working on the tool under partly unexplained circumstances in May 2014 [1]. As early as 2013, VeraCrypt was created as a fork of TrueCrypt [2]; it provides the same functionality while eliminating most known bugs from TrueCrypt.

Whereas VeraCrypt is designed for encrypting large datasets, TruPax [3] is a smaller tool and a good alternative for quickly and reliably keeping small amounts of data safe from unauthorized users. The volume format is compatible with VeraCrypt, so the volumes you create can be opened and edited in either application.

Setting up TruPax

Although the program has been in development for some time, you will so far not yet find TruPax in the software repositories of the popular distributions; this means picking up the program from the project page. Versions for 32-bit and 64-bit Linux are available along with the source code.

The source archive is a roughly 40MB ZIP file, which you can unpack and install in any directory. In the newly created subdirectory, call the `./install.sh` script, which copies the complete program to `/opt/trupax/` and adds a starter to the application menu. Because the installation script temporarily requires administrator privileges, it prompts you for a password. If you work in a distribution that only supports `sudo` after manually installing the appropriate software package, then you need to move the entire program folder to the `/opt/` directory yourself.

After completing the setup, you can launch the GUI version of the program with the `trupaxgui` command. The `trupax` command-line tool, which also exists, accepts a large number of parameters, which it outputs when called without options. To use the software more conveniently in the future without changing to the command line, you can create a starter in the menu tree after the manual installation.

Interface

The application window (Figure 1) appears spartan at first glance: On the left is a large, empty list area immediately after launching the program. On the right are some buttons that let you create, edit, or unzip volumes and add some storage options. At the bottom left is a color-inverted activity and progress bar that indicates the current state of the software.

At the top is a small horizontal menubar with File and Help entries. If you have any doubt about a feature, simply mouse over the item in question: The program provides bubble help that briefly explains the operation in a sentence or so.

Packing

The first step is to define which files or folders you want TruPax to add to the volume and encrypt. To do so, click *Add Files* or *Add Folder* at the top right in the program window. If you want TruPax to encrypt directories recursively (e.g., to add all the subfolders to the volume), check the *Include Subfolders* option.

If you want the software to save complete paths as in the source tree without saving the

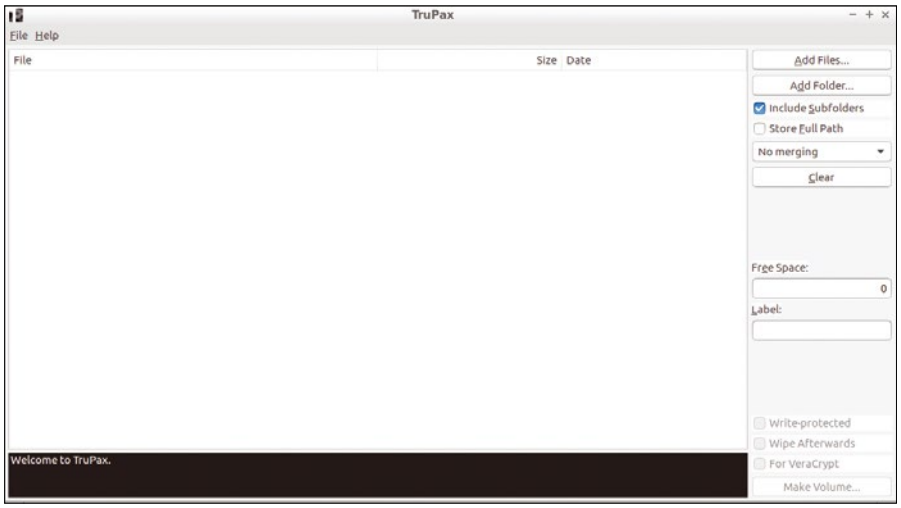


Figure 1: Spartan but yet very easy to use: the application window in TruPax 9.

drive identifier, then also check the *Save Full Path* option. As soon as you click on the dialog to add files or folders, a file manager appears to let you select the desired content. The software displays the volume size and the size of the selected files in the status area at the bottom of the program window.

Finally, enter a name for the volume in the Label box. The name can comprise numbers and letters and up to 15 characters. Do not confuse this identifier with the actual name of the volume file: The identifier serves as a kind of drive identifier for the VeraCrypt volume.

Same Name

On systems with large amounts of data, files commonly have the same names within nested directory hierarchies. If you save such a complete folder hierarchy in a volume in TruPax without saving the source tree, you will inevitably experience conflicts when you save the files with identical names.

To prevent TruPax accidentally overwriting the original file with a newer version, select the *No merging* option in the corresponding drop-down menu of the program window. If this entry reads *Merge* instead, a newer file with the same name will overwrite the older file, and the program does not take upper- and lowercase into account. The option *Merge Case Sensitively* only overwrites an older file with a newer file of the same name if the spelling of the two file names matches exactly.

Before creating the volume, you check the *Write-protected* option on the right in the program window to tell the program to make the volume read-only, and thus immutable. The option *Wipe Afterwards* deletes the original files as soon as TruPax has created the

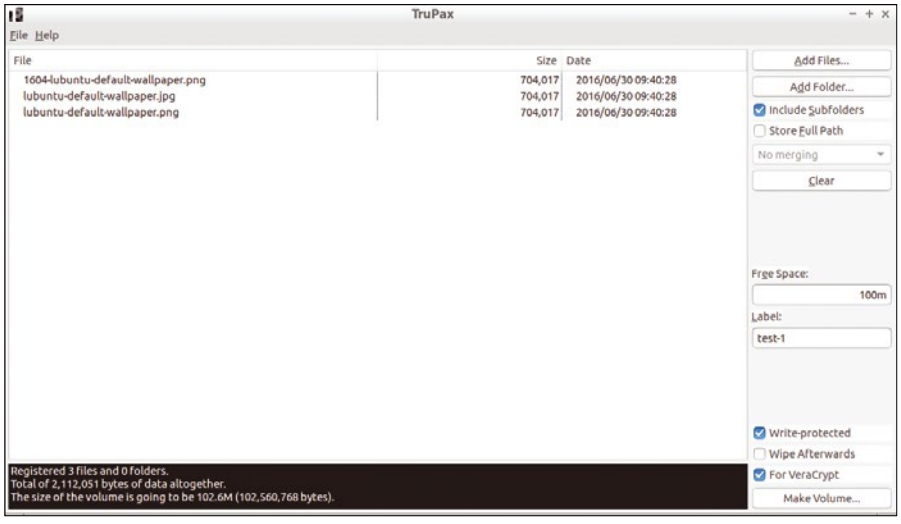


Figure 2: Let TruPax create the volume after selecting the data to be encrypted.

Figure 3: TruPax lets you irrevocably delete volumes.

volume. To keep the volume compatible with VeraCrypt, if necessary, check the *For VeraCrypt* parameter.

XXL

TruPax automatically sets the size of the volume so that it matches the total volume of the files and folder of the original content. The program thus avoids wasting disk space on the target medium. However, if you add files to the volume that you want to edit later, it is advisable to allocate additional space to the volume.

In the *Free Space* field, type an integer value followed by the letter *k*, *m*, or *g* for kilo-, mega-, or gigabytes. TruPax adds the desired storage capacity and displays the total in the status display in the lower left corner.

After completing the file and folder selection, click the *Make Volume* button at the bottom (Figure 2). In the dialog box that then appears, enter the name for the volume file and select the storage path. TruPax typically uses the TrueCrypt-compatible volume format as the file extension (`.tc`). If you enable VeraCrypt compatibility, the volume name ends with `.hc` instead.

In the next step, the routine prompts you for a password that you must specify later on when you open the volume. Select a password as secure as possible so the volume is not exposed to the risk of dictionary attacks; note that TruPax is case sensitive. You need to enter the password a second time to confirm.

After assigning the password and clicking on *Proceed*, TruPax writes the volume to the hard drive. The application uses a 256-bit key with the secure AES algorithm for encryption. Depending on the data being encrypted, the process can take a while to complete.

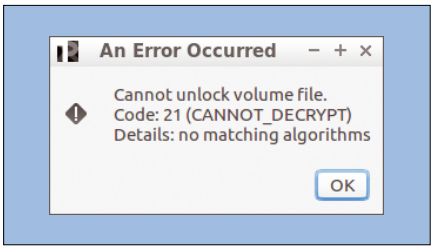


Figure 4: No way back: Invalid volumes do not reveal data.

Unpack and Destroy

To unpack volumes again, select *File* | *Extract*. TruPax opens a dialog in which you specify the destination for the unpacked directories and files and the volume to extract. Next, just enter the password for the volume and wait for the software to extract the data to the desired location. The original volume is kept.

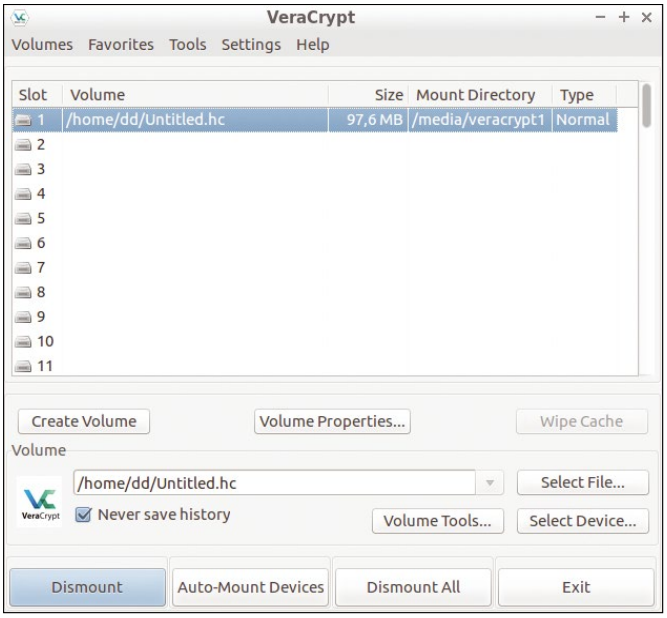


Figure 5: You can use TruPax volumes on systems with VeraCrypt.

Unneeded volumes represent a potential security risk, as long as you do not dispose of them safely. A simple delete does not reliably remove a volume from the disk; it can be reconstructed just like any other file by special software. To make an obsolete volume physically disappear from the respective data carrier, TruPax offers a function for destroying files.

The option for this is *File* | *Invalidate* | *Continue*, which destroys the header in the volume and thus throws away the key. In the corresponding deletion dialog, you will also find the *Continue and Delete* option, which physically removes the volume from the disk so that the space can be used for other purposes (Figure 3).

Invalidated volumes that are not deleted simply remain on the system. However, you no longer have a way to access the data stored in the volume, even if you know the assigned password. As soon as you try to extract such a volume in TruPax, the software outputs an error after you enter the password (Figure 4).

In VeraCrypt, you can easily mount a volume encrypted with TruPax as a conventional volume, assuming you prepared it appropriately by setting the *For VeraCrypt* option in the TruPax program window. If you then load such a volume in VeraCrypt, the TrueCrypt successor decrypts the volume and mounts the data as a drive (Figure 5). The name entered as the Label in TruPax is used as the volume label. A volume that is modified in VeraCrypt can be opened again in TruPax.

Conclusions

TruPax gives you the ultimate tool for encrypting smaller datasets that you want to access on the go. The software impresses with its fast and stable functions and its intuitive interface that eliminates visual overkill. The 256-bit AES encryption feature reliably backs up the data and safeguards it from prying eyes.

Moreover, TruPax is compatible with VeraCrypt, which makes it much more flexible than a proprietary volume format, so you can access your data on computers that only have VeraCrypt in place – even if it's a Mac or Windows machine. TruPax is thus recommended without restrictions for smaller datasets on removable storage. ■■■

INFO

[1] True Goodbye: 'Using TrueCrypt Is Not Secure', by Brian Krebs: *https://krebsonsecurity.com/2014/05/true-goodbye-using-truecrypt-is-not-secure/*

[2] VeraCrypt: *https://veracrypt.codeplex.com*

[3] TruPax: *https://www.coderslagoon.com/*

Detecting spam users automatically with a neural network

SPAM STOPPER

Build a neural network that uncovers spam websites. *By Chris Hinze*

Website builders – online hosting services that provide tools for non-technical users to build their own websites – are frequently exploited by spammers looking for a convenient launching pad. Checking thousands, or sometimes millions, of web pages manually to look for evidence of a spammer is both tedious and inefficient.

In this article, I show how to build a suitable spam-searching neural network with help from Google's TensorFlow machine learning library [2] [3] and TFLearn [4], a library with a high-level API for TensorFlow. Even if you don't spend your days searching for spammers, the techniques described in this article will give you some insights on how to harness the power of neural networks for other complex problems.

AUTHOR

Chris Hinze studies IT at the University of Erlangen-Nuremberg, Germany, and works at Benjamin Lochmann New Media GmbH as a web developer. His work there involves back ends for smartphone apps, and he has collaborated on automated spam recognition for *homepage-baukasten.de*.

Training Day

The neural network needs both positive and negative samples in order to learn. This solution starts with a manually compiled list of sample users divided into spammers and legitimate users, taking care to distribute both types in equal numbers. Alongside this classification (spammer or not spammer), the data set contained the user's name or the website that belongs to the user, the IP address with which the site is registered, and the language version associated with the site.

As a result of the solution described here, the neural network now automatically recognizes new spammers as they register. The next step is to combine this automatic check with a manual check. A Python script automatically blocks sites that the network classifies with a very high probability of being spam, and an employee manually checks the sites that are deemed high probability.

Sound Network

Neural networks are mathematical models that can approximate any function. A neural network is guided by networked neurons similar to those in the human brain, such as in the visual cortex. What makes these networks special is that you do not have to model their behavior explicitly; instead, you train the network using sample data.

Neural networks help out when it is difficult to model functions manually, and they are often used in image and speech recognition. You need to provide the neural network with training data that has already been classified, and it will then attempt to classify new data in a similar way.

A single artificial neuron comprises several weighted inputs and an activation function, which is usually non-linear and helps to determine the output

value of the neuron. There is also a threshold value or bias, which complements the weighted inputs, thus influencing the activation function. The mathematical formula behind this concept is as follows:

The formula uses the w vector to weight the input vector x and calculate the sum of both. It then adds the bias b, using the activation function phi. When developers skillfully combine several neurons, they can compute more complex functions (see the box titled "Solving Problems with Neural Networks").

SOLVING PROBLEMS WITH NEURAL NETWORKS

A single neuron can already solve linearly separable problems. The binary OR function is an example of such a linearly separable problem. If you enter the possible inputs in a coordinate system, the two output values can be separated with a straight line (the top right neuron in Figure 1).

Few problems, however, are so easy to solve. A single neuron is typically not enough to make a classification. Full networks composed of neurons are used in practice, because more complex challenges can largely be split into separable sub-problems, which individual neurons can then solve.

Figure 1 shows the binary exclusive OR, which proves not to be linearly separable. A single line is not sufficient to separate the two ones from the zeros. As the propositional logic is aware, the XOR function consists of a combination of two conjunctions:

Both the two conjunctions and the disjunction can in turn be separated linearly. It is therefore possible to model the binary exclusive OR with three neurons, with one of them receiving the outputs of the two others. This combination of neurons forms a small, two-layered neural network.

As the small example demonstrates, deep learning experts can calculate complex functions with ease by combining multiple neurons. The strength of a neural network increases with the number of layers used. The layers allow experts to compute more functions.

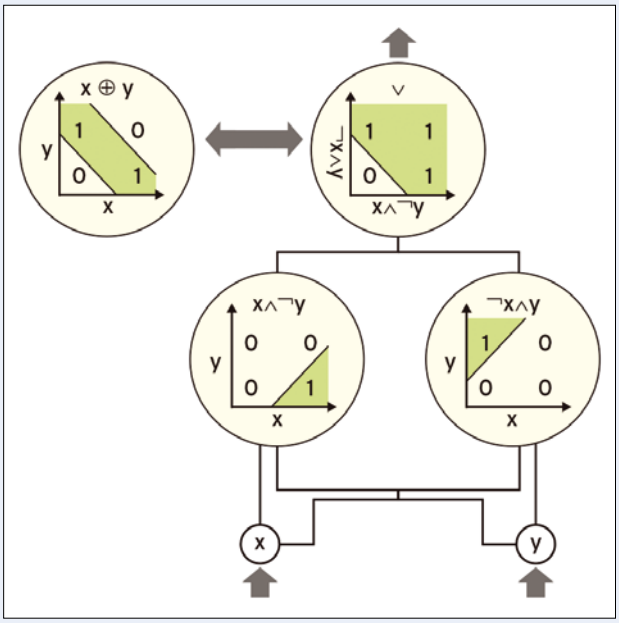


Figure 1: A binary XOR is not linearly separable per se, although it can be split into linearly separable sub-problems.

Networked Learning

Several layers of interconnected neurons form a neural network (Figure 2). These layers consist of at least an input layer, which receives the input values, and an output layer, on which the data arrives after passing through several hidden layers. All the neurons on a particular layer generally use the same activation function.

Neural networks learn through an optimization process that determines the parameters of the network, the weightings of the connections, and the bias of all the neurons, then refines these values step by step. The process that determines parameters is one of the optimization problems. This process involves the use of many traditional numerical analysis methods (e.g., the gradient method [5]).

The script first initializes the network using random parameters. Next, the script applies the training data set to the neural network and determines the difference between the network's results and the correct results from the training data. The gap between these results is the loss, which the script attempts to minimize in the course of the optimization process.

Circuit Training

The training is conducted iteratively. During a pass (an *epoch*), the developer

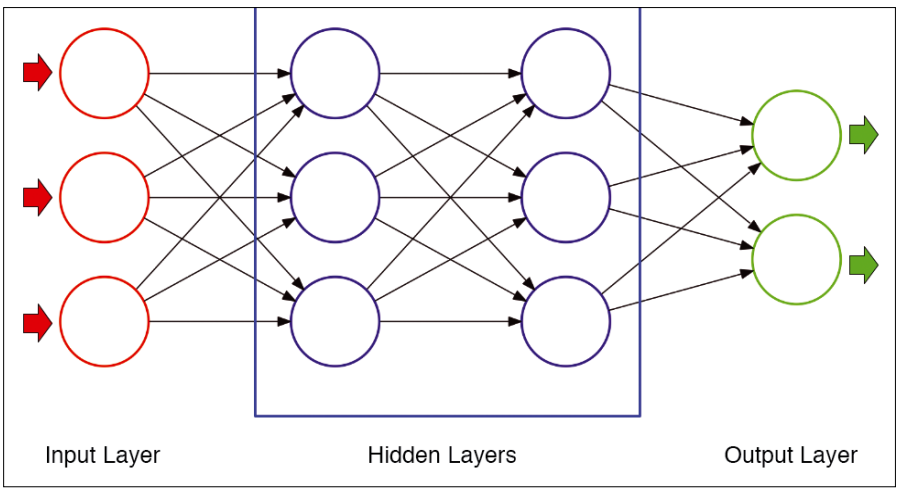


Figure 2: A neural network with two hidden layers.

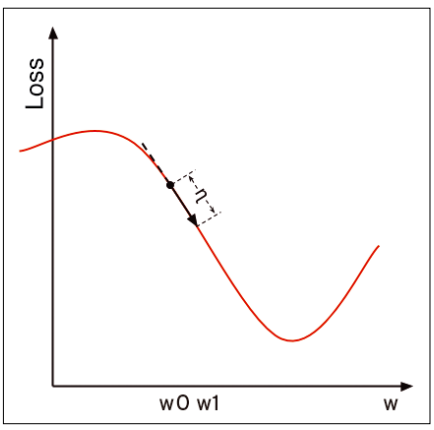


Figure 3: The gradient descent method determines a function's minimum in this simplified depiction.

passes the complete data set through the neural network one time and works out the loss. You do not use all the data at once in this process but, rather, divide it into smaller portions (batches) for performance reasons.

In the course of a later epoch, the training script sets new parameters via the optimization process, incurring a smaller loss. Ideally, the loss will converge toward a specific minimum value. The network can then be regarded as trained, and the training script stores the calculated weightings and threshold values that produced the minimum loss.

Figure 3 shows a simplified view of how the gradient descent method determines the minimum for an individual weighting `w`. The *x*-axis is the weighting, and the *y*-axis is the loss function's value with this weighting. The training script determines the loss graph's differentiation – and thus its slope – with each iteration of the gradient method at the point of the current weight, then moves a step in the learning rate (eta) direction.

Interesting Properties

The fields of the data sets, which are used as inputs for the network, are known as properties. The neural network works with real numbers, which means names and IP addresses cannot be added directly as properties in the form of strings.

Our experience shows that spammers often use very cryptic usernames. We were able to derive the following properties to help identify spammers: the length, the number of hyphens, the number of numerals, the differentiation of the characters, the number of vowels, the number of non-letters, and the occurrence of certain keywords (e.g., *credits*, *100mg*, *taler*).

A geolocation database breaks down the country, matching a particular IP address and the ISP. The on-hand data reveals how often an ISP operates as a spammer, how frequently a combination of a particular country of origin and chosen language appears for the website builder, and which countries transmit an especially large amount of spam.

The next step is to sort out properties that do not correlate strongly with the class and thus contribute little to the outcome. The reason for sorting out the data that doesn't correlate strongly is that a smaller network can be trained more quickly and needs fewer resources. I can use a correlation matrix to discern how well-suited properties are for spam detection.

Listing 1 shows a Python script for setting up the correlation matrix. The script reads a CSV file with the data, computes the correlation matrix using the `np.corrcoef()` function, and finally

LISTING 1: *correlation.py*

```
01 import csv
02 import numpy as np
03 import os
04 import sys
05 import dnn
06 import tflearn
07 from PIL import Image, ImageDraw
08
09 def read_file(filename):
10         data = []
11         with open(filename, "r") as file:
12                 reader = csv.reader(file, delimiter=",")
13                 next(reader) # Skip header
14                 for row in reader:
15                         row.pop(0) # Remove username
16                         row = map(float, row)
17                         data.append(row)
18         return data
19
20 def calculate_correlation_matrix(data):
21         with np.errstate(invalid="ignore"):
22                 return np.corrcoef(data, rowvar=0)
23
24 def draw_matrix(matrix, filename, size=20):
25         n = len(matrix)
26
27         img = Image.new("RGB", (n*size, n*size))
28         draw = ImageDraw.Draw(img)
29
30         for i in range(0, n):
31                 for j in range(0, n):
32                         color = "hsl(0, 0%%, %d%%)" % (matrix[i][j]*100)
33                         draw.rectangle((i*size, j*size, (i+1)*size,
                                        (j+1)*size), fill=color)
34
35         img.save(filename)
36
37 if __name__ == "__main__":
38         if len(sys.argv) < 2:
39                 sys.exit("Usage: python correlation.py <input-csv-file>")
40
41         input_file = sys.argv[1]
42
43         data = read_file(input_file)
44         matrix = calculate_correlation_matrix(data)
45         draw_matrix(np.absolute(matrix), "correlation.png")
46
47         print(matrix)
```

generates a PNG file with the density plot of the matrix. The script ignores the first column (in the username sample data) during this process. If the CSV file contains other values that are not real numbers, you will have to modify the `read_file()` function accordingly. The class, which distinguishes spammers from legitimate website builders, is intended to be in the last column.

The density plot (Figure 4) shows an overview of which properties are particularly suitable. Each row and column cover a property. The lighter the field, the higher the correlation between the row property and the column property. The last row and column reveals the correlation with the class. For this reason, the lighter colored the field in the last row and column, the better suited the properties to the classification.

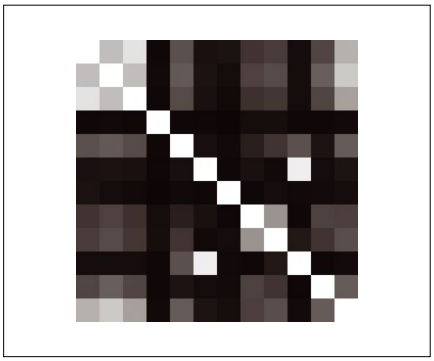


Figure 4: The correlation matrix density plot reveals the interdependencies among properties.

The correlation matrix also reveals whether two characteristics are excessively similar and whether it would be sufficient to classify them as one where possible. The properties *6* (the number count in the username) and *10* (the number of non-letters) would be an example of this. The white field indicates a strong relationship between both these variables. It is therefore sufficient to take property *6* into account, because *10* provides no additional information.

Hyperparameters

The lion's share of work with neural networks is in determining the structure or configuration of the network with the aid of hyperparameters. Developers usually perfect this process manually by individually training each network configuration and comparing the results until ending up at a good configuration. Hyperparameters include the number and size of the layers, the activation functions for the layers, the number of epochs, the size of the data batches, the optimization process, and the learning rate.

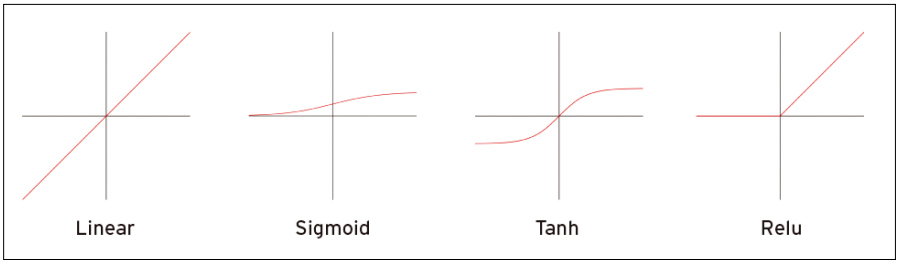


Figure 5: The most important activation functions at a glance.

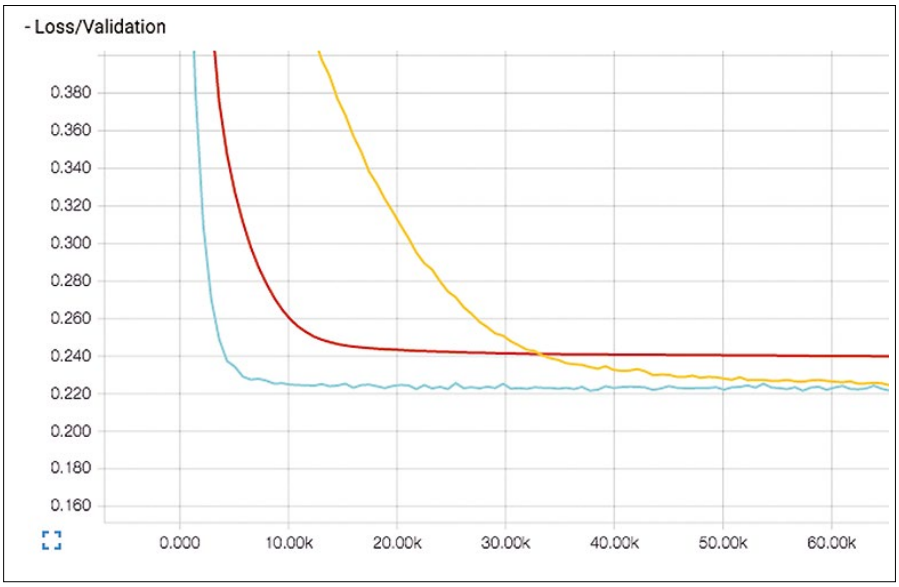


Figure 6: Loss during training at different learning rates, with the blue curve representing the ideal.

TFLearn offers a variety of activation functions in the *tflearn.optimizations* package. Figure 5 depicts the most important of these functions; the easiest is the identity or the linear activation function, which returns the input value unaltered. The sigmoid function is non-linear and so is more interesting as an activation function than the linear equivalent. The function is restricted and only produces positive values between 0 and 1.

Tanh can be compared with sigmoid, except that it returns values between -1 and 1. A further activation function is known as a rectified linear unit, or Relu. You can think of this as a linear function with a threshold value. Relu converges very quickly and is the recommended function at this time. We also use it for our network.

Another important activation function goes by the name softmax. The softmax function creates a relationship between the value of the neuron and the values of other neurons in the layer. Its special characteristic is that all output values in this layer add up to 1. Users often use this function for the output layer in networks whose purpose is to classify. The network's output can then be interpreted as probabilities for the individual classes.

You also pick an optimization process along with the layers and their activation functions. Adam, an algorithm with which you can generally achieve good results, is often used by developers to train neural networks as an alternative to the classic gradient method. Adam also needs a learning rate. The preset value of 0.001 is a suitable learning rate to start with, although it can be reduced to achieve an even better outcome where possible. All the optimization techniques supplied with TFLearn are included in the *tflearn.optimizers* package.

You can train your network and measure its accuracy by using the hyperparameters you have found, before proceeding to vary the parameters. You continue to repeat this process until you can no longer significantly increase the accuracy.

Figure 6 shows the loss during training in graph form. The graph will indicate whether the learning rate is too high or low. The loss graph is intended as far as possible to resemble a falling exponential curve (the blue graph). If the learning rate is too high, the loss initially drops quickly, although it is possible that it may converge prematurely (red). This indicates that you have not yet found the optimum. The loss only drops very slowly when the learning rate is too low, and it is more likely that you actually find the global optimum (yellow). You can increase the batch size if the loss graph is too noisy.

Genuine Configuration

Although the optimization process aims for a good result with use of the training data, it is possible that the network could, in a sense, learn the training data by heart instead of generalizing and developing a common method. This effect is called overfitting [6]. To test whether the network does generalize, you have to split the data set in advance into training data and validation data. Use the training data for optimizing parameters, determine the accuracy via validation data still unknown to the network, then compare the different network configurations.

In addition to training and validation data, you'll also need to set aside some data for testing the neural network. Training, validation, and test data is typically distributed at an 80:10:10 ratio.

The network here consists of a layer of 30 neurons with the linear activation function and four Relu layers with 30 neurons each. Adam optimizes with a 0.00001 learning rate. After 500 epochs, the activation function achieves an accuracy of 92.6 percent with the validation data and still manages 91.9 percent with test data. The training lasted around 30 minutes on a MacBook Pro.

You can normalize the data in advance if you want to produce a better result. To normalize, begin by weighting

LISTING 2: *train* Function from *dnn.py*

```
01 [...]
02 def train(name, dir, input, net, optimizer, batch_size, epochs):
03         # Import and distribute data
04         data, labels = parse_csv(*input)
05         X_train, X_val, X_test, y_train, y_val, y_test = split_data(data, labels)
06
07         # Preprocessing: zero-center, and normalize
08         X_train, mean = tflearn.data_utils.featurewise_zero_center(X_train)
09         X_train, std = tflearn.data_utils.featurewise_std_normalization(X_train)
10         X_val = tflearn.data_utils.featurewise_zero_center(X_val, mean)
11         X_val = tflearn.data_utils.featurewise_std_normalization(X_val, std)
12         np.save(dir+"mean", mean)
13         np.save(dir+"std", std)
14
15         # Training
16         net = tflearn.regression(net, optimizer=optimizer)
17         model = tflearn.DNN(net, best_checkpoint_path=dir)
18         model.fit(X_train, y_train, validation_set=(X_val, y_val),
                     show_metric=True, n_epoch=epochs, batch_size=batch_size, run_
                     id=name)
19 [...]
```

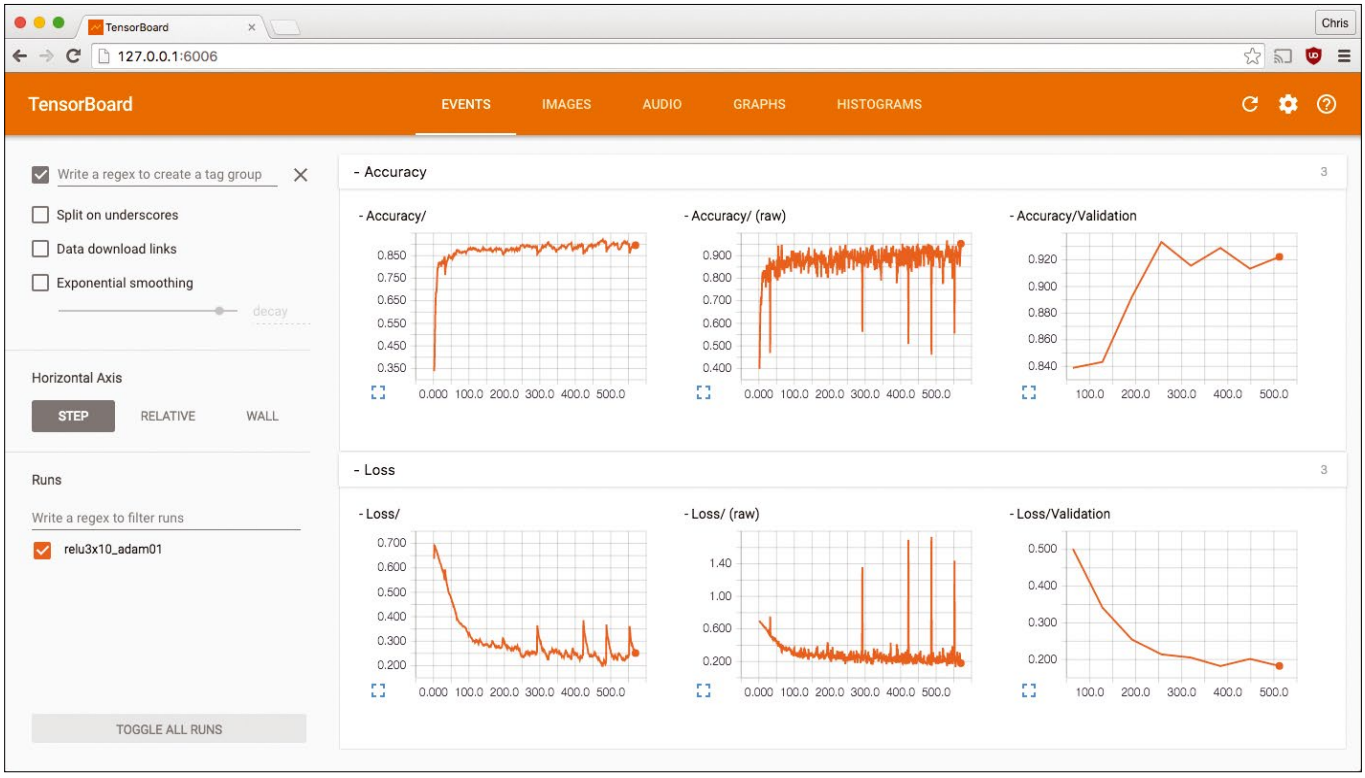


Figure 7: TensorBoard graphically processes the statistical data from TensorFlow.

the values to the average by subtracting the arithmetic mean. You then bring the data to the interval [-1, 1] by dividing by the standard deviation. It is important to determine the arithmetic mean and standard deviation only with the training data and to reuse these values for the validation, testing, and production data.

TensorFlow

The TensorFlow library implements the neural network. TensorFlow comes with TensorBoard, a web app that displays statistics from TensorFlow and processes them graphically (Figure 7). TensorBoard helps with optimizing the network structure and is very well suited for monitoring the learning process and comparing different configurations. TFLearn also uses TensorFlow and provides easier-to-use functions for neural networks.

See the instructions at the TensorFlow project website for information on how to install TensorFlow [7]. Python package manager `pip` then installs TFLearn on the hard drive. You will need at least version 0.2.2. We installed TFLearn via Git, as described at the project website [8]. You will also need the `numpy`, `sklearn`, `scipy`, and `pillow` packages.

Listing 2 shows excerpts from the `dnn.py` script, which trains and evaluates neural networks (Listing 3) and is used for classification (see Listing 4).

Listing 2 demonstrates how the developer can train a network with TFLearn. The script first loads the input data from the CSV file before splitting and normalizing the data. The script then saves the mean value and standard deviation of the training data to apply them to the production data later on. The `best_checkpoint_path` parameter (line 17) instructs TFLearn to save the state of the network each time as soon as the script achieves higher accuracy with the validation data during training. The script trains the network across epochs with `n_epoch` via the `model.fit()` method from the TFLearn library.

Listing 3 deals with evaluation of the network. First, the code re-imports the CSV file with the input data. The script divides the data equally with each execution according to the random number generator's fixed seed. It then retrieves the mean value and the standard deviation, normalizing the test data. To finish, it evaluates the network status with the best accuracy and determines the hit rate of the test data set with `model.evaluate()`.

The code in Listing 4 allows the network to classify the data. It also reads the data to be classified as a CSV file, normalizes it, loads the network status, and classifies data sets with `model.predict()`. The function outputs the probabilities for each class per data set.

LISTING 3: *evaluate* Function from *dnn.py*

```
01 [...]
02 def evaluate(name, dir, input, net, optimizer, batch_size, epochs):
03  [...]
04          # Import and distribute data
05          data, labels = parse_csv(*input)
06          X_train, X_val, X_test, y_train, y_val,
                  y_test = split_data(data, labels)
07
08          # Preprocessing: zero-center, and normalize
09          mean, std = np.load(dir+"mean.npy"), np.load(dir+"std.npy")
10          X_test = tflearn.data_utils.featurewise_zero_center(X_test, mean)
11          X_test = tflearn.data_utils.featurewise_std_normalization(X_test, std)
12
13          net = tflearn.regression(net, optimizer=optimizer)
14          model = tflearn.DNN(net, best_checkpoint_path=dir)
15          model.load(get_checkpoint(dir))
16
17          print(model.evaluate(X_test, y_test))
18  [...]
```

LISTING 4: *predict* Function from *dnn.py*

```
01  [...]
02 def predict(name, dir, input, net, optimizer, batch_size, epochs):
03  [...]
04          # Import data
05          X, _ = parse_csv(sys.argv[3], input[1])
06
07          # Preprocessing: zero-center, and normalize
08          mean, std = np.load(dir+"mean.npy"), np.load(dir+"std.npy")
09          X = tflearn.data_utils.featurewise_zero_center(X, mean)
10          X = tflearn.data_utils.featurewise_std_normalization(X, std)
11
12          net = tflearn.regression(net, optimizer=optimizer)
13          model = tflearn.DNN(net, best_checkpoint_path=dir)
14          model.load(get_checkpoint(dir))
15
16          for line in model.predict(X):
17                  print(line)
18  [...]
```

Experiments

The `experiments.py` file determines the structure and configuration of the individual networks (Listing 5). An experiment is a function. It returns a tuple consisting of input data, network, optimization methods, batch size, and number of epochs. Here, `input` is a tuple including the CSV file name, the list of columns with properties, and the column that specifies the class.

The methods `tflearn.input_data()`, `tflearn.batch_normalization()`, and `tflearn.fully_connected()` define the network's layers. The developer indi-

vidually assigns each layer a size and an activation function. The input layer receives the properties, with their size depending on their number. There should be a batch normalization layer ahead of each fully connected layer. The softmax type, with the number of classes as its size, is recommended for the last layer.

The batch size reveals how many data sets the optimization method processes at once during an iteration. The higher the figure, the more smoothly the loss and accuracy graphs are plotted in TensorBoard and the faster TensorFlow trains the network, although a higher batch size also requires more resources. A figure between `128` and `1024` is a good reference value.

The `epochs` variable in the script represents the number of epochs, so it specifies how frequently the neural network manages to learn the entire training set. This network mostly converged after 500 iterations, so it is worth using a higher value in this instance. You can manually cancel the training as soon as the loss no longer changes.

Call up:

```
python dnn.py train experiment
```

to train a network, where `experiment` is the name of a function that you created earlier in the `experiments.py` file. The current loss and the accuracy of both the training and the validation data appear directly in the terminal during training. TensorBoard represents the values as a graph. To start TensorBoard, use:

```
tensorboard -logdir=/tmp/tflearn_logs/
```

Usually you call the web interface in the browser with *http://127.0.0.1:6006*. The plotted graphs for accuracy and loss are hidden behind the *Events* entry. You can select the experiments under the Runs section. The data usually updates automatically every 120 seconds; you can also update manually on request by clicking the button at the top right.

If you have found a good configuration and feel satisfied with the accuracy of the network, you can perform a check by entering

```
python dnn.py evaluate experiment
```

to establish how well the network detects still-unknown test data. You must also store new data sets in a CSV file to classify them and call them up with the command:

```
python dnn.py predict experiment Input
```

The network classifies the data and outputs the probabilities for the corresponding classes line by line. In the example here, the first value for each line describes the probability of the user having good intentions, and the second provides the same for spammer suspects.

Future

The method described in this article has some limitations. Although a neural network can come close to any complex function, it may be the case that the optimization processes do not produce the optimum solution. In this case, the network only achieves a low accuracy level.

A further potential problem is caused by unbalanced or contradictory training data, which, for instance, might quite accidentally involve only the spammers having hyphens in their names. There is also the previously mentioned risk in large networks of overfitting, where the network learns the training data by heart but doesn't gain the ability to evaluate new, unknown data.

Despite these limitations, you can check far more pages than before using the method described in this article, because the neural network pre-sorts potential spammers. If additional spammers are found manually, you can feed them into the network later in the form of training data. ■■■

LISTING 5: *experiments.py*

```
01 import tflearn
02
03 def relu3x10_adam01():
04         input = ("trainingdata.csv", ("name_length", "char_diff",
                    "hyphens", "numbers", "vocals"), "spammer")
05
06         net = tflearn.input_data(shape=[None, 5])
07         for i in range(3):
08                 net = tflearn.batch_normalization(net)
09                 net = tflearn.fully_connected(net, 10, activation="relu")
10         net = tflearn.batch_normalization(net)
11         net = tflearn.fully_connected(net, 2, activation="softmax")
12
13         optimizer = tflearn.optimizers.Adam(learning_rate=0.001)
14         batch_size = 128
15         epochs = 300
16
17         return input, net, optimizer, batch_size, epochs
```

INFO

[1] All listings for the article: *http://www.linux-magazin.de/static/listings/magazin/2016/12/machine_learning/*

[2] TensorFlow: *https://www.tensorflow.org*

[3] TensorFlow: Large-scale machine learning on heterogeneous systems? (2015): *http://download.tensorflow.org/paper/whitepaper2015.pdf*

[4] TFLearn: *http://tflearn.org*

[5] Bengio, Yoshua, Practical recommendations for gradient-based training of deep architectures. *In* G. Montavon, G.B. Orr, and K.-R. Müller (eds.), *Neural Networks: Tricks of the Trade*, 2nd ed. Springer-Verlag, 2012, pp. 437-478

[6] Overfitting: *https://www.ibm.com/developerworks/community/blogs/jfp/entry/Overfitting_In_Machine_Learning*

[7] Installing TensorFlow: *https://www.tensorflow.org/versions/r0.10/get_started/os_setup.html#pip-installation*

[8] Installing TFLearn: *http://tflearn.org/installation/*

Klaus Knopper answers your Linux questions

Ask Klaus!

By Klaus Knopper

KLAUS KNOPPER

Klaus Knopper is an engineer, creator of Knoppix, and co-founder of LinuxTag expo. He works as a regular professor at the University of Applied Sciences, Kaiserslautern, Germany. If you have a configuration problem, or if you just want to learn more about how Linux works, send your questions to: *klaus@linux-magazine.com*

Android USB Backup

I would like to back up my Android-based smartphone to a large hard disk attached to a Linux computer while the phone is connected via USB. Apart from all the apps out there that use their own archive format, is there something simple that I can do from the Linux desktop for creating a backup that is also readable from Linux without needing a special program or app?

The universal tool for accessing Android at the system level over USB or the network is the `adb` command, which exists in the *android-tools-adb* Debian and Ubuntu package.

Install `adb` (use the "unstable" Debian branch to get the newest version) with:

```
sudo apt-get install [-t unstable] ➥
  android-tools-adb
```

`adb` has a way to create Android backup archives of apps and associated data via the `backup` command, but it also allows you to execute shell commands on the phone. If the Android system has the option to run commands as root, a full backup via the standard `tar` command is possible, otherwise only globally readable data, such as recorded pictures or videos, can be archived.

To access the smartphone, `adb` must be enabled in Android's "developer options" first. By default, this menu is hidden and only appears if you click seven times on the *Build number* item in Android's *Settings* | *About device* menu. After seven clicks, a popup should tell you that "You are now a developer!" or "Developer mode has been enabled." Back to the main Settings menu, the box under *Developer options* | *USB debugging* should be checked to allow you to access the phone over USB (Figure 1).

If after connecting the phone to the computer via USB cable a notification requests your permission to access the phone from that computer, tick the checkbox and click *OK*. In the Terminal window on the Linux desktop side, you can now start the `adb` commands for accessing the phone. Table 1 shows a small summary of commands.

The `exec-in` and `exec-out` feature, which is not mentioned in the man page or in help yet, provides an easy and fast way to transfer binary data between Android and the attached Linux computer. After entering

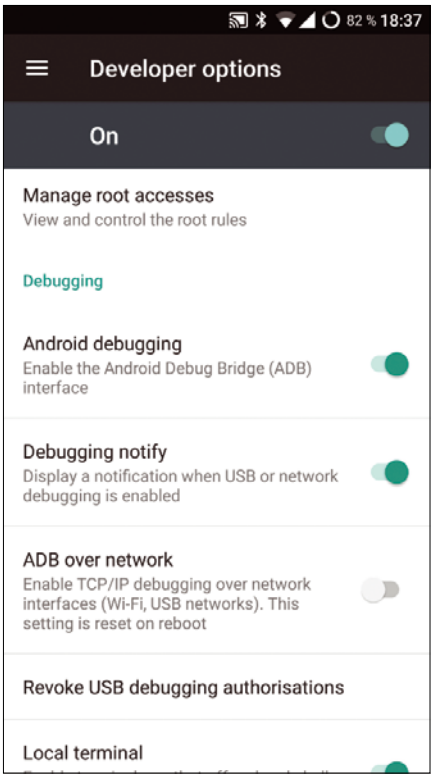


Figure 1: The enabled Developer options screen.

```
adb root
```

(works only for rooted phones), you can create a compressed tar stream of the phone's critical data partition. To transfer and store it to a backup file on your computer, issue:

```
adb exec-out ➥
  "GZIP=-1 tar -zcpf - /data ➥
   2>/dev/null" > android-data.tar.gz
```

- `GZIP=-1 tar -zcpf -` creates (`-c`) and streams a tar archive to standard output (`-f -`), which is compressed (`-z`) with GZIP level 1 (`-1`) for fastest compression, while also archiving the secure Linux contexts and special permissions (`-p`). Secure Linux contexts, by the way, are important; they allow apps to access their specific data files, because Android in "strict" SELinux mode will in fact deny access to files if these special modes are not set correctly, causing apps to crash. Of course, it's a security feature, so `tar` on newer Android versions has the (also yet undocumented) `-p` option for storing additional permission information.
- `/data` is the partition containing settings and app data. (The stuff you definitely want to keep, even if you decide to upgrade the `/system` partition with a new Android version.)
- `2>/dev/null` discards error messages. If the command terminates very quickly, you might want to remove that option to see what happens (maybe the `-p` option is not supported in older Android versions), but error messages and notices can pollute the tar archive data if inserted into the stream, so it's recommended to send them to `/dev/null` if the command is otherwise OK.
- `> android-data.tar.gz` sends the captured tar archive stream from standard output to the file `android-data.tar.gz`. No intermediate data is stored on the Android side.

To check the archive content, run:

```
tar -ztvf android-data.tar.gz
```

(No `adb` here; it's local!) To restore parts of the archive back to the phone, the command is:

```
adb exec-in ➥
  "tar -zxpf - data/media/0/Music" ➥
  < android-data.tar.gz
```

- `tar -zxpvf -` decompresses (`-z`), extracts (`-x`), and restores data from the archive streamed as input (`-f -`), also restoring secure Linux contexts (`-p`).
- `data/media/0/Music` indicates the only part of the archive you want to restore (e.g., the `Music` directory here). Unless you are recovering from a crash or after a "factory reset," you probably don't want to restore the entire archive, because newer versions of settings and files would be overwritten. The default (without specifying a directory) is to extract the entire archive.
- `< android-data.tar.gz` feeds the stored archive as an input stream to `adb`.

For transferring single files or directories from and to Android, the commands

```
adb pull <filepath>
adb push <filepath>
```

are also useful, but they are not as flexible as `tar`. ■■■

TABLE 1: adb Commands

Command	Description
`adb root`	Give root permissions for subsequent commands on the smartphone (requires a rooted phone and may ask for permission).
`adb shell`	Log in to a remote shell on the smartphone; feels almost like a shell on a desktop computer (`cd`, `ls`, ..., type `exit` to leave).
`adb backup <options>`	Back up apps and app data in encrypted format.
`adb restore <options>`	Restore apps and app data in encrypted format.
`adb exec-out "<command>"`	Run command on Android and capture its unfiltered binary output (stdout) – undocumented feature.
`adb exec-in "<command>"`	Run command on Android and feed the unfiltered console input (stdin) – undocumented feature.
`adb help`	Short help (man page is more verbose).

Mail-merged party invitations with gLabels

Party On

Add an individual touch to invitations or cards to help your event start with a bang.

By Mario Blättermann

A grab bag of software can still be found in electronics stores alongside last season's tax return software. Illustrious products such as greeting card creators and the like accompany jaded graphic and video suites and font packages. Although you will find hardly any counterparts to these titles in the free software world, you do have some more intelligent alternatives.

COMPILING GLABELS

If you are building gLabels from the source code [2], you use the usual three-step process of `configure`, `make`, and `make install`. In addition to GTK 3.x, you also need Libxml and Libsvg – all with the appropriate developer packages – as well as the XML:Parser Perl module. Other extras, such as the Zint or QRencode libraries for generating barcodes, are optional. Depending on your system's GTK configuration, the configuration script might complain about other missing libraries – but nothing too exotic – so the installation is unlikely to fail, except on BSD systems that don't get along well with GTK3.

When the task is creating invitations and place cards for an event, some Linux users might wish they had some of these Windows programs. They make life easy, with ready-made design templates into which you only have to enter names. With a little ingenuity, though, you can just as easily use gLabels [1].

First published more than 15 years ago and updated constantly, the first version of this program was intended for designing and printing labels and business cards like those available commercially. However, just as most people can manage to knock a nail into the wall with a pair of pliers, if need be, gLabels will let you create totally different and no less professional-looking printed materials. Although, gLabels is not a mature desktop publishing program like Scribus, the opportunities it offers are fine for the task at hand, and the learning curve is pleasantly flat.

The program can be found in the repositories of mainstream distributions. The current version 3.4.0 was released in April 2016, so it is likely available in the repositories – or, you can build gLabels from source (see the "Compiling gLabels" box). The mail merge function has been around for years, so the task at hand can be completed even with far older versions of the software.

Ready, Set, Print

After starting the program, go to the *File | New* menu and select the template that meets your needs. For this example, you can choose a label or card template freely from the collection.

The *Generic* brand in the left drop-down menu (Figure 1) contains a few common paper formats, such as the A6 format, which acts as the template in this example. After clicking *Next*, select the orientation (*Normal* or *Rotated*, i.e., portrait or landscape) and begin to unleash your ideas for the design.

Enabling the grid in the background (*View | Grid*) generally makes it easier to arrange individual objects. First, you need text, such as: "You are cordially invited to Tom and Diana's New Year's Eve party at 7:00pm." With the mail merge feature, you can invite many other guests without repeatedly entering the salutation and names, which greatly simplifies the process.

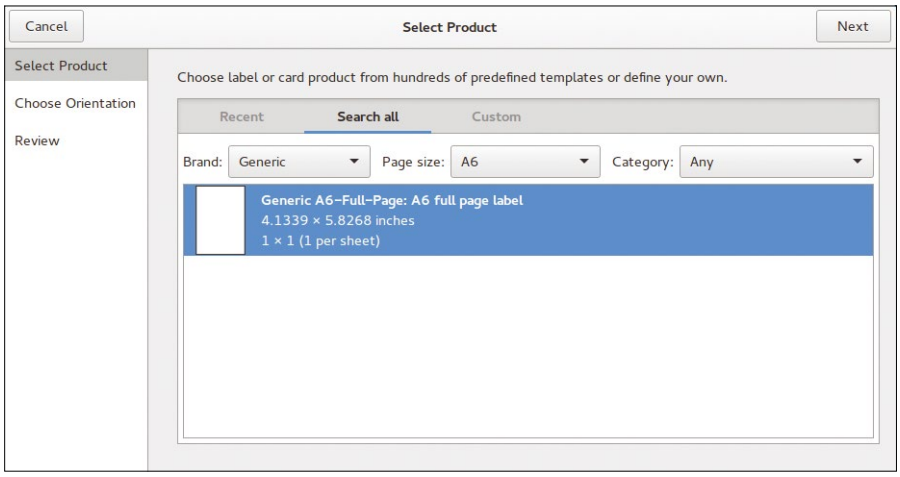


Figure 1: The generic templates include paper formats, typically in ISO or US standards.

You can store the names of your guests in an external text file in the comma-separated value () format, which you can export from a spreadsheet, or you can even create the list in a simple text editor. Write each salutation and the corresponding names, separated by a comma, one after another in rows (Figure 2). The comma acts as the delimiter; the application accepts spaces in the rows without problem.

Fresh Text

Now you can insert the text into the drawing area in gLabels by clicking the *T* in the toolbar or by accessing *Objects* | *Create* | *Text* through the menubar. Clicking on the drawing area lets you create a suitable object whose properties you can edit in the panel on the right.

To size the text frame properly, drag the green handles, and to move the text box, mouse over the border between the green handles and press and hold the left mouse button while dragging.

The text itself needs a little more attention: A decorative font in an appealing size, possibly centered on the page, adds a more aesthetic touch to your invitation. You will find the appropriate settings in the *Style* tab in the properties panel. Another interesting option here is the *Shadow* tab, where you can add a shadow in a freely selectable color and offset to the text. The software does not blur the shadow, so you should use the effects sparingly. Too little contrast and too much offset could make the text illegible. The *x* and *y* offsets can be adjusted in a granular way in increments of thousandths of an inch or tenths of a millimeter; adjusting the Opacity also improves readability.

You can insert objects, such as images, lines, and borders with or without a fill color to your heart's content. All of these objects can be accessed directly from the toolbar. Useful graphics and images are also available in the public domain (e.g., check out the Openclipart project [3]).

If you are a little too bold in your placement of objects and lose sight of one, remember that it's not really gone – just covered up by other objects, because each object has its own layer. You can bring a layer to the front or send it to the back in the *Objects* | *Order* menu. Figure 3 shows what the example looks like with a color-filled box in the background, a shaded and colored font, and a small graphic.

Batch Processing

This draft of the invitation only takes care of one invitation. To incorporate mail merge for personalized invitations, open the *Objects* | *Merge properties* menu (Figure 4). As the Format, select CSV; you may notice that the menu supports quite a few more formats, such as TSV, which uses a tab as the delimiter. For Location, specify the previously created CSV file. You could even access your Evolution address book, although you would need to add a salutation.

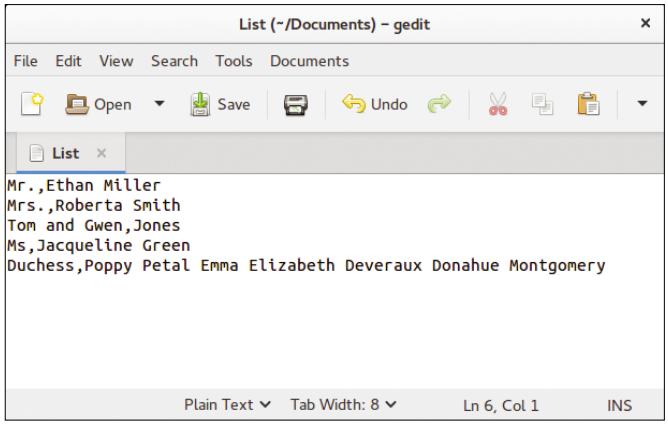


Figure 2: gLabels processes the data for your guests' names and addresses from lines of text in a simple format.

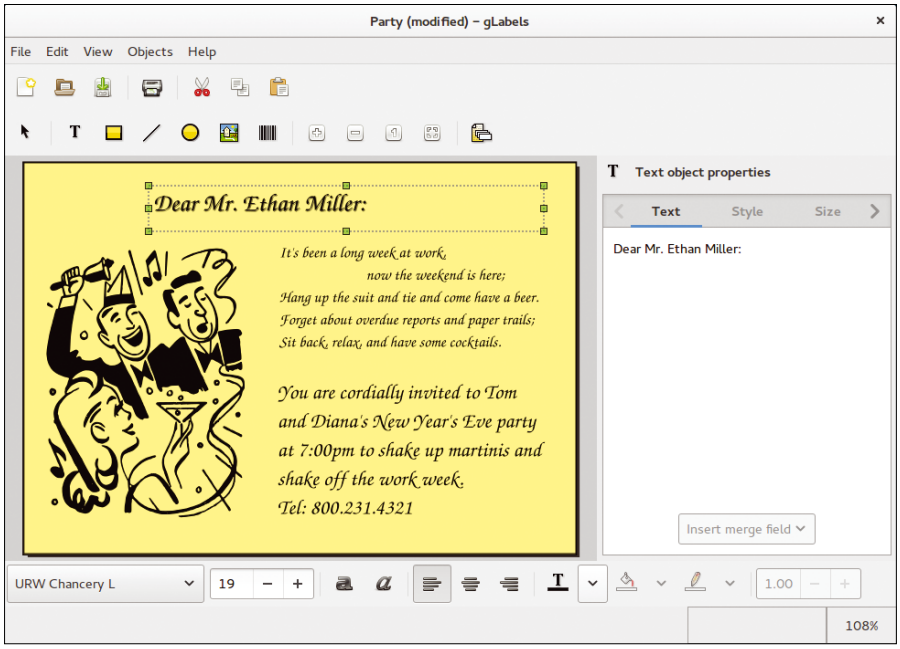


Figure 3: The invitation offers some potential for improvement, but it isn't bad for a first draft.

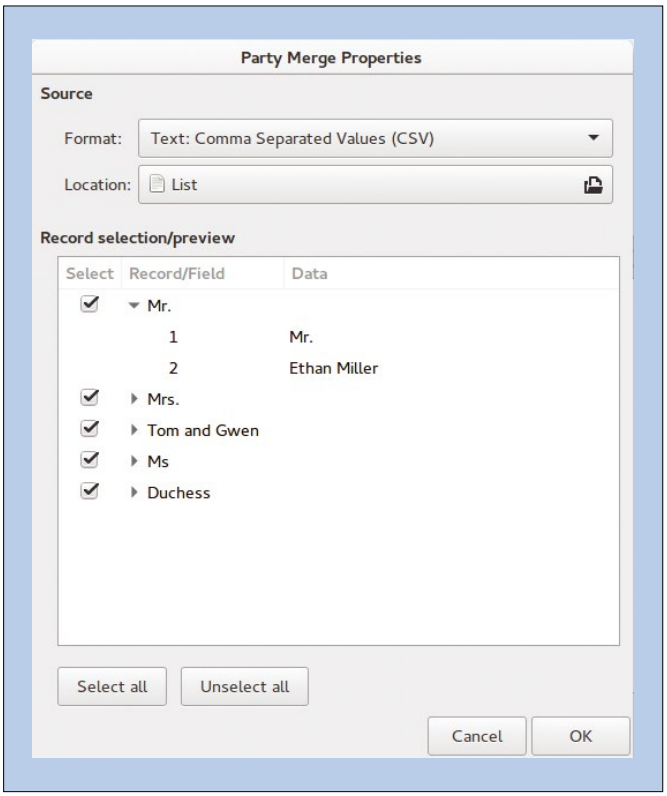


Figure 4: Setting up the name list for processing in your mail merge document.

In the properties panel in the main window, a previously inactive *Insert merge field* option is now ready for use, so you only need to replace the names in the text with variables. Two entries appear: one for the salutation and one for the name. Figure 5 shows what the text looks like after this step.

Now it's time for a first print preview. Select the printer icon in the toolbar and press the *Preview* button in the resulting window. Your variables are of variable length, so you can't be sure how much space the text will occupy in the salutation box. Figure 6 clearly shows how the nobility still requires special attention today.

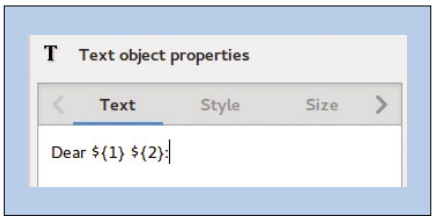


Figure 5: The inserted variables pave the way for batch processing of entries from the CSV file.

Dinner Is Served!

The invitations have been sent out, and now it's time to turn to the place cards. gLabels does not let you manage multiple templates in a project, even if they are on the same sheet of paper. Thus, you need to select *File | New* to open a new template. To create a folding card, you can use the generic A6 template again and rework it, or you can opt for a smaller template (e.g., a business card). The standard format results in a visible area of approximately 3.5x1 inches (85x24mm) when folded and positioned next to the place setting. In the design phase you should mark the center of the template with a line, which is used as an edge for folding and can be deleted from the finished layout.

You again need a CSV file as a data source, although it is simpler than the one for the invitations; this time, you only need to write the names themselves in a text file. You could even rehash the design from the invitations to impress your guests, who will suspect that you called in a professional service provider.

Now embed the objects for text and images as described before. People are inquisitive by nature; often guests furtively check the silverware for authenticity or try to discover where you sourced the place cards. You could add a copyright note on the back of the card to make your product look even more professional – and satisfy your guests' curiosity without them having to be intrusive about it.

To create the copyright line, open a text object in the drawing area. To avoid the text being printed upside down, rotate it in 90-degree increments with the appropriate function in the *Objects | Rotate/Flip* menu. After rotating twice clockwise or counterclockwise, the text – now standing on its head – will regain its feet when you fold and place the card.

Figure 7 shows the finished place card, as shown in the gLabels drawing area. It's time for another detailed check in the print preview. In the print dialog,

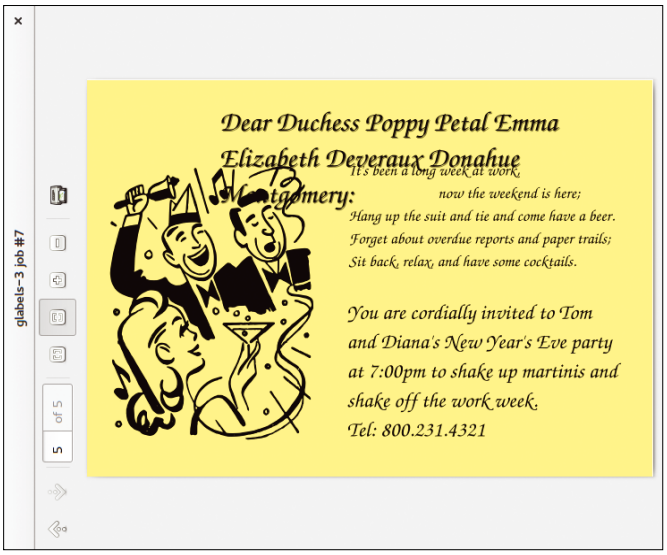


Figure 6: An excessively long name can mess up the layout, so you should always inspect the print preview.

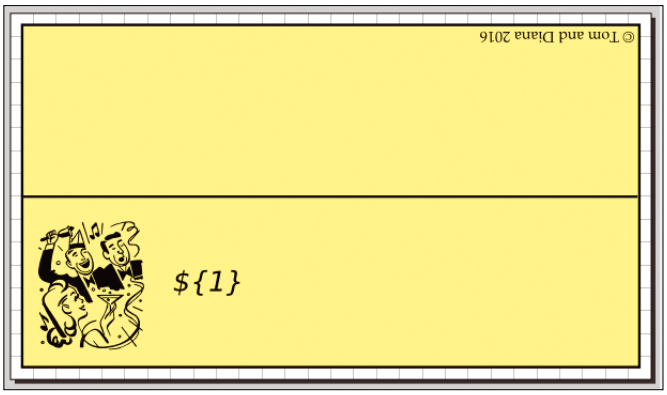


Figure 7: gLabels shows you the place card with the variable instead of names.

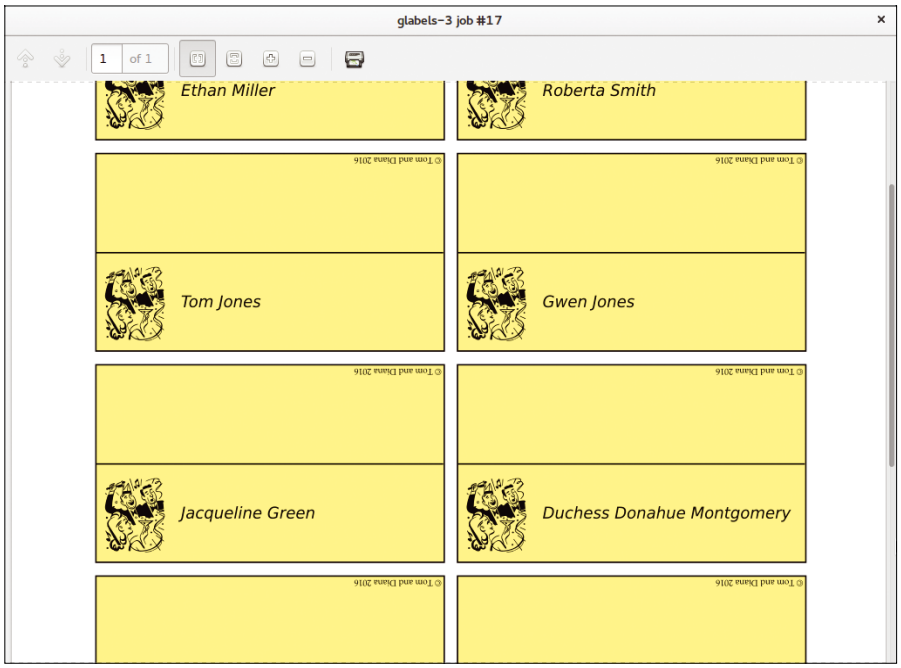


Figure 8: The print preview lets you know whether your design works.

you will want to enable the option *Print borders (to test print alignment)* in the *Labels* tab. This is recommended, especially if you have no frame around the layout, to show the borders of each card.

The preview shows that expanding the variables to real names worked (Figure 8). You now only need to delete all the guidelines and print the place cards.

Conclusions

Comparing the use of gLabels with the use of pliers for knocking in nails turned out to be pretty unfair; after all, gLabels mastered the task with flying colors. It is only one example of many; you can coax much more out of the software with a little creativity and surprisingly little effort.

gLabels lets you print invitations and place cards with a made-to-order layout and professional design. The software provides less functionality than a desktop publishing program but features mail merge. Creativity is still needed, but with an invitation of your own design and matching place cards, at least you will be unlikely to hear your guests saying: "Seen that, done that." ■■■

INFO

[1] gLabels: *http://glabels.org*

[2] Latest gLabels version: *http://ftp.gnome.org/pub/GNOME/sources/glabels/3.4/glabels-3.4.0.tar.xz*

[3] Openclipart: *https://openclipart.org*

The sys admin's daily grind: httpstat

My Point of View

Httpstat is a special stopwatch you can use to discover how long web servers take to serve up a static or dynamic HTML page. Visible performance lags indicate optimization potential for the server. *By Charly Kühnast*

Httpstat is a Python script that wraps itself around cURL. Apart from Python 2 or 3 and cURL, it has no other dependencies. You can retrieve it from the GitHub repository and call it using:

```
wget https://raw.githubusercontent.com/
  reorx/httpstat/master/httpstat.py
python httpstat.py <URL>
```

If the Python installer `pip` is present on your system, you also can pick up the script and call it with:

```
pip install httpstat
httpstat <URL>
```

Although you can leave an `http://` out of the URL, you cannot omit `https://` for web pages secured with TLS.

Figure 1 shows httpstat measuring an unencrypted call. Four milliseconds for a DNS reply is a really good value, but I cheated: The name of the site is cached on my local Dnsmasq. As soon as my computer has to turn to my provider's DNS, the value rises to 80-200ms. The TCP handshake is 22ms, which is about par for the course.

The time the server needs to create the page (*Server Processing*) shows whether the web server has some tuning potential that I have not tapped. My example is not representative, because instead of HTML, the server simply outputs *301 Moved Permanently*, which means I should have called the page using HTTPS. A browser would do that independently, but not cURL.

Figure 2 requests the same page using HTTPS. The lookup and TCP values remain the same, but the *TLS Handshake* takes forever for this static page. The value can go up to several seconds for a big site with a large volume of dynamic content and advertising banners.

Httpstat is not controllable using command-line parameters because they would be fielded by cURL; however, you can influence the tool with environment variables. The line

```
export HTTPSTAT_SHOW_SPEED=true
```

tells httpstat to show how quickly the web page is delivered (e.g., *speed_download: 219.6 KiB/s, speed_upload: 0.0 KiB/s*). The httpstat website [1] explains all of the variables and has links to the httpstat implementation in Go, Bash, and PHP. ■■■

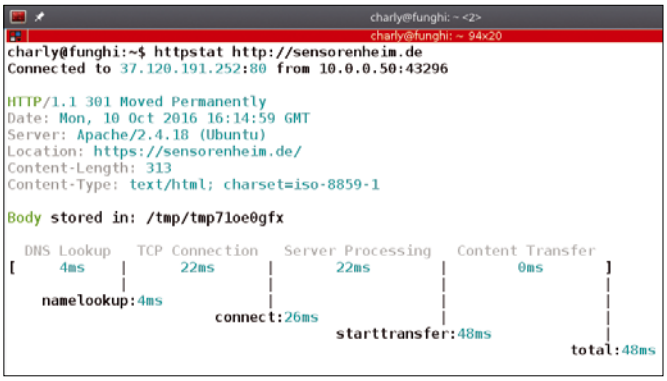


Figure 1: The page that httpstat requests via HTTP, and receives quickly, is only an error message in reality.

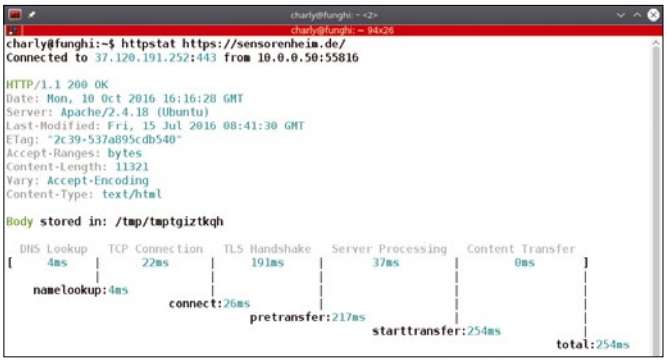


Figure 2: When requested using HTTPS, the HTML page obviously takes longer to appear.

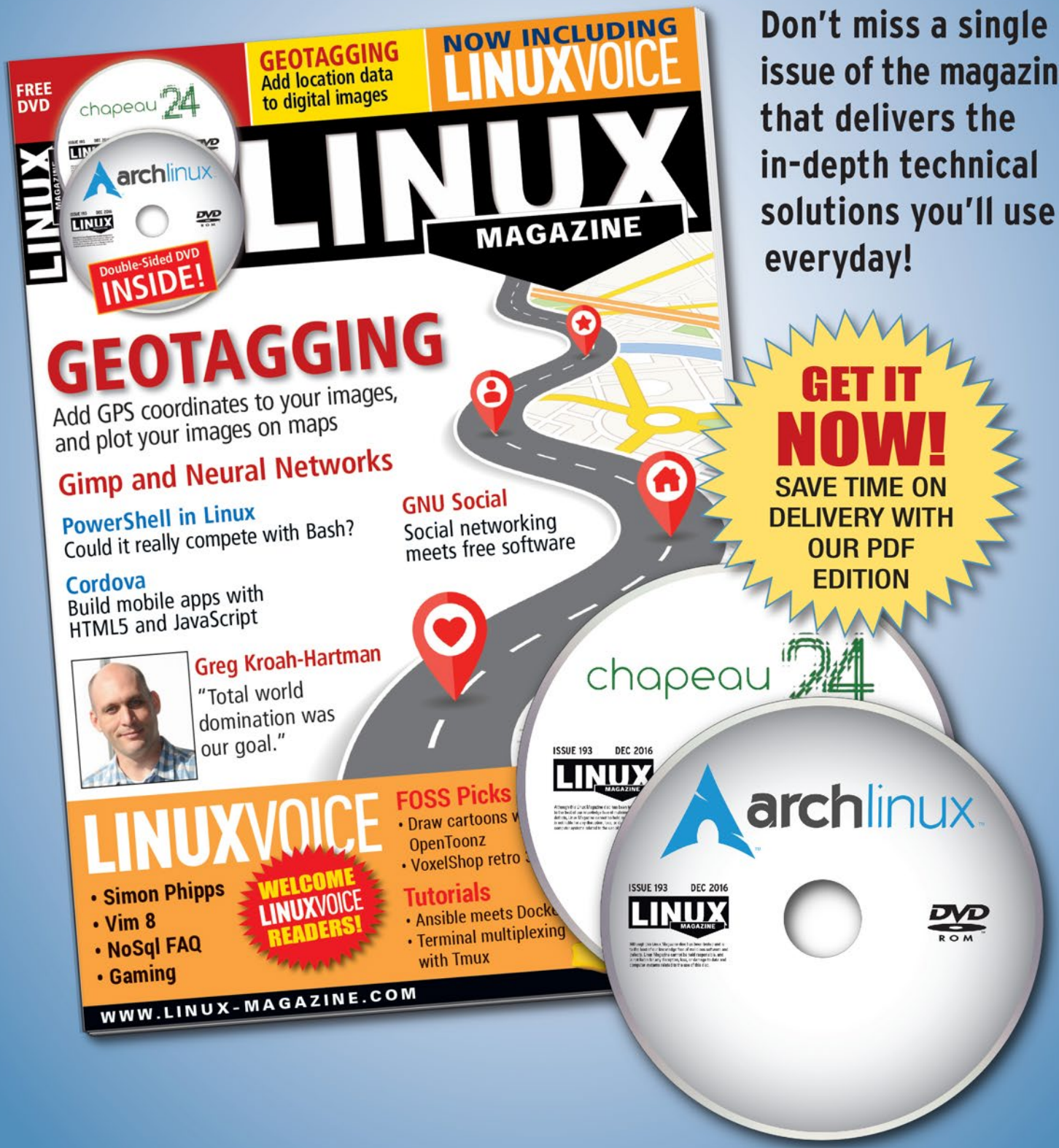
INFO

[1] httpstat: *https://github.com/reorx/httpstat*

CHARLY KÜHNAST

Charly Kühnast manages Unix systems in the data center in the Lower Rhine region of Germany. His responsibilities include ensuring the security and availability of firewalls and the DMZ.

Subscribe now!



Don't miss a single issue of the magazine that delivers the in-depth technical solutions you'll use everyday!

shop.linuxnewmedia.com/subs

Time-saving preview of surveillance videos

City View

Rather than stare at boring surveillance videos, in which nothing happens 90 percent of the time, Mike Schilli tries the OpenCV image recognition software, which automatically extracts the most exciting action sequences.

By Mike Schilli

In my home city of San Francisco, hardly a day goes by without hundreds of cars, garages, and homes being broken into. Instead of getting upset about this, I tend not to keep anything of value in easily accessible places, and I have also installed security cameras so that I can peruse the video footage of thieves at work for my personal amusement.

Wireless, Even

Of course, installing a security camera is no easy task, because you need to install a cable and route it to the monitor. Although the camera itself often communicates wirelessly with the control panel, it still needs a power supply, and a power supply is not easy to come by in hotspots such as the underground parking lot or the stairwell.

Recently, a company called Arlo started to sell child-fist-sized, battery-powered cameras [1], which amateur detectives can simply hang up using a magnet (Figure 1). These pocket wonders wirelessly send recorded videos to a hub at a distance of up to about 100 feet, which in turn sends the data via the Internet to a server, from which a variety of smartphone apps or a website transfers the data to the user's screen on request.

Conservative Operation

To reduce the load on the four lithium batteries powering the camera so they can last for about one month, the camera is allowed to wake up about a half a dozen times a day if it detects motion in its vicinity and then transfer a one-minute video. You then download the movie from Arlo's website (Figure 2) to see thieves, say, dragging your new bike out of the garage. You typically only see motion at the beginning of a surveillance video, the rest of the one-minute footage normally shows nothing but motionless background (Figure 3).

MIKE SCHILLI

Mike Schilli works as a software engineer in the San Francisco Bay Area. He can be contacted at *mschilli@perlmeister.com*. Mike's homepage can be found at *http://perlmeister.com*.

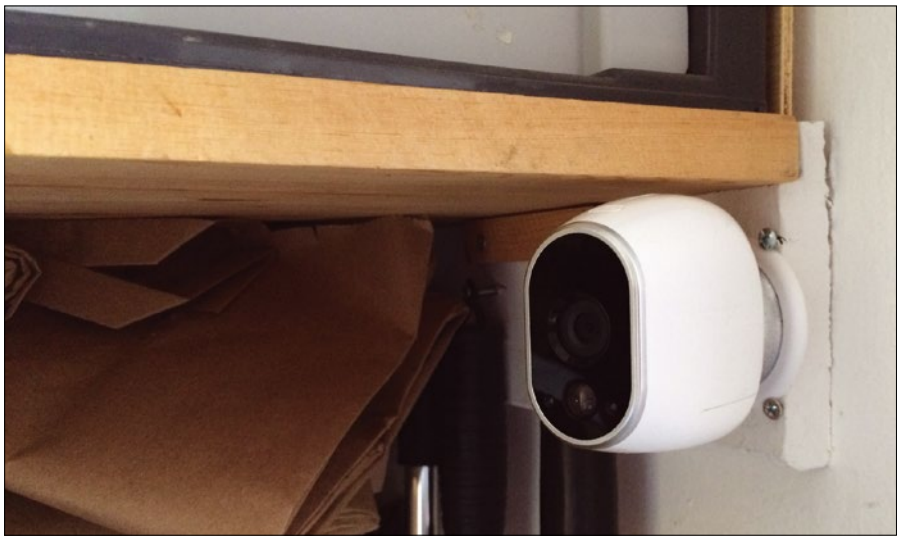


Figure 1: The pocket-sized Arlo camera fits into the smallest cubbyhole and needs no power supply.

Fast Forward to the Action

"Cut to the chase," people say when someone fails to come to the point. This probably refers to action films, where the audience does not want to see long-winded, suspense-building scenes but prefers to fast-forward to the car pursuit at the climax of the Hollywood production.

In this sense, it would be nice for the software to scour the video for frames in which a subject actually moves through the scene, so that the viewer knows whether the videos are worth watching and, if so, the location to which to fast forward in the video.

Jerky Action

Fast-motion playback of the movie, for example, with the help of the `fps` (frames per second) parameter in MPlayer, would be the easiest option:

```
mplayer -framedrop -fps 150 video.mp4
```

The `-framedrop` parameter simply throws away frames if the CPU fails to keep pace when decoding. The result is a movie that runs roughly five times as fast as normal, which looks like something from the early days of movie making, like the old hand-cranked takes of Charlie Chaplin.

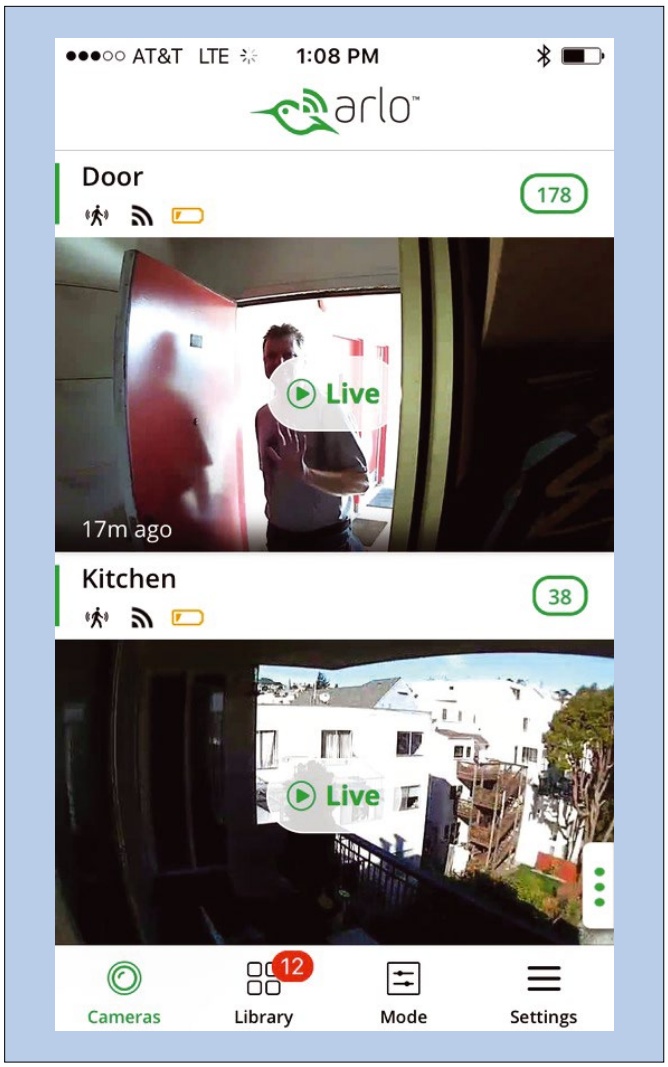


Figure 2: Videos recorded by the camera are available for download.

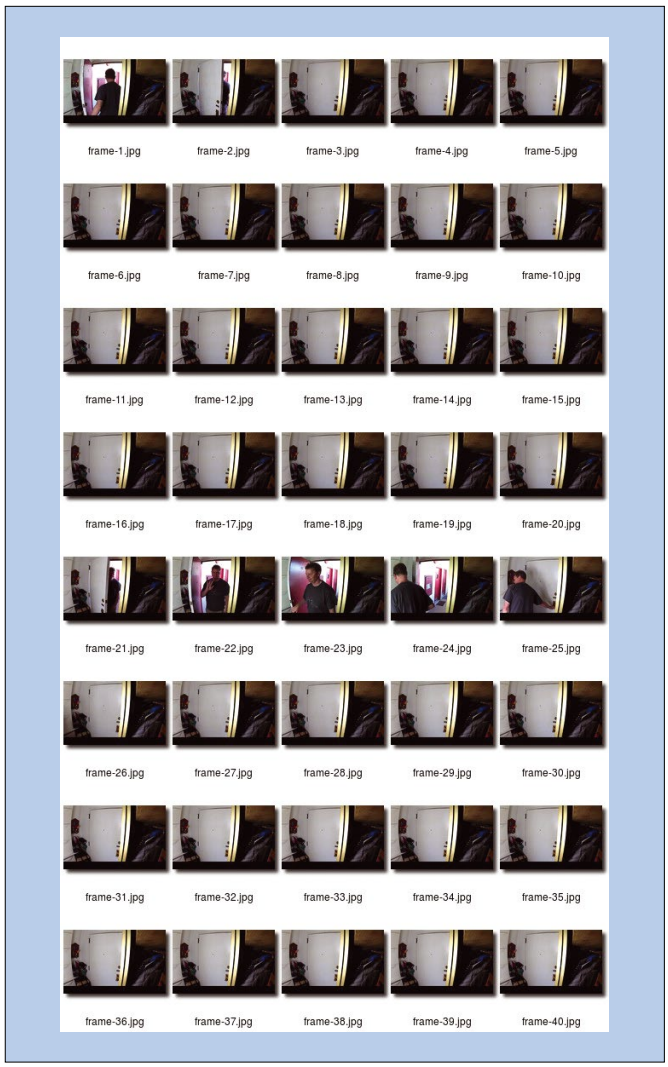


Figure 3: Most frames only show the closed door.

This procedure requires a human reviewer, whereas I was looking for an automated process that lists the most important seconds of the retrieved video as metadata, along with illustrative thumbnails, much like a photographer's contact sheet.

Motion Detection

Much has happened in the field of pattern recognition in image processing in the last few years, and the OpenCV [2] package even offers some highly scientific routines as an open source program. To determine whether someone or something is moving through the video, a program needs to read the frames in the video stream and determine which pixels have shifted from the (x, y) coordinates to the $(x + \Delta x, y + \Delta y)$ coordinates. If you find a large contiguous area for which this is true, then you can assume that a movement took place between two frames.

One of the procedures included with OpenCV is called Lucas-Kanade [3]; it attempts to find optical flow – areas around central points that jointly move from one frame to the next – in a video. To do so, it first determines a number of interesting areas, where monitoring promises success and that another algorithm can extract from an image by focusing on image points in areas with a recognizable structure or on the edges of objects.

To perform the Lucas-Kanade (LK) analysis, the OpenCV package (on Ubuntu, this is *libopencv-dev*) provides

LISTING 1: max-movement-lk.cpp

```
01 #include "opencv2/opencv.hpp"
02
03 using namespace std;
04 using namespace cv;
05
06 const int MAX_FEATURES = 500;
07 const int MAX_MOVEMENT = 100;
08
09 int move_test(Mat& oframe, Mat& frame) {
10     // Select features for optical flow
11   vector<Point2f> ofeatures;
12   goodFeaturesToTrack(oframe,
13     ofeatures, MAX_FEATURES, 0.1, 0.2 );
14
15     // Parameters for LK
16   vector<Point2f> new_features;
17   vector<uchar> status;
18   vector<float> err;
19   TermCriteria criteria(TermCriteria::COUNT
20       | TermCriteria::EPS, 20, 0.03);
21   Size window(10,10);
22   int max_level   = 3;
23   int flags       = 0;
24   double min_eigT = 0.004;
25
26     // Lucas-Kanade method
27   calcOpticalFlowPyrLK(oframe, frame,
28     ofeatures, new_features, status, err,
29     window, max_level, criteria, flags,
30     min_eigT );
31
32   double max_move = 0;
33   double movement = 0;
34   for(int i=0; i<ofeatures.size(); i++) {
35     Point pointA
36       (ofeatures[i].x, ofeatures[i].y);
37     Point pointB
38       (new_features[i].x, new_features[i].y);
39
40     movement = norm(pointA-pointB);
41     if(movement > max_move)
42       max_move = movement;
43   }
44   return max_move > MAX_MOVEMENT;
45 }
46
47 int main(int argc, char *argv[]) {
48   int i = 0;
49   Mat frame;
50   Mat oframe;
51
52   if (argc != 2) {
53     cout << "USAGE: <cmd> <file_in>\n";
54     return -1;
55   }
56
57   VideoCapture vid(argv[1]);
58   if (!vid.isOpened()) {
59     cout << "Video corrupt\n";
60     return -1;
61   }
62
63   int fps = (int)vid.get(CV_CAP_PROP_FPS);
64
65   i++;
66   if(!vid.read(oframe)) return 1;
67
68   cvtColor(oframe, oframe, COLOR_BGR2GRAY);
69
70   while (1) {
71     if (!vid.read(frame))
72       break;
73     i++;
74
75     cvtColor(frame,frame,COLOR_BGR2GRAY);
76     if(move_test(oframe, frame))
77       cout << i/fps << "\n";
78     oframe = frame;
79   }
80
81   return 0;
82 }
```

the `calcOpticalFlowPyrLK()` function with no fewer than 11 parameters.

Acrobatics with Cmake

The C++ program in Listing 1 [4] reads in a video file and detects any movement of objects between frames. Converting it into an executable program requires some compilation acrobatics with `include` files and link libraries; your easiest approach here is to use `cmake` and its meta Makefile in Listing 2.

Typing the `cmake .` command (the dot stands for the current directory, in which the `CMakeLists.txt` file resides) followed by the `make` command starts the lengthy compilation process, which finally produces the `max-movement-lk` binary.

LISTING 2: CMakeLists.txt

```
1 cmake_minimum_required(VERSION 2.8)
2 project( max-movement-lk )
3 find_package( OpenCV REQUIRED )
4 add_executable( max-movement-lk max-movement-lk.cpp )
5 target_link_libraries( max-movement-lk ${OpenCV_LIBS} )
```

It expects a video file and outputs movie location values in seconds for scenes containing motion.

To do this, the `main()` function reads the video file name from the command line and starts `VideoCapture` provided by the OpenCV package in line 57. The frame rate is read from the video file in line 63 and stored in the `fps` variable. Because the LK algorithm works best with grayscale images, lines 68 and 75 bleed the color out of every frame to be analyzed.

A `while` loop iterates across all the frames, and the `move_test()` function checks in line 76 whether any motion occurred between the last frame read, `oframe`, and the current `frame`. If so, line 77 divides the counter value by the FPS value of the video and thus computes the time at which the motion occurred in the video as a value in seconds.

Me-Too Algorithm

The algorithm borrowed from *OpenCV Essentials* [2] uses the `move_test()` function (lines 9-45) to call the OpenCV `goodFeaturesToTrack()` function (line 12) to detect points of interest in the old frame (`oframe`); the maximum number points of interest is limited to 500 by the constant `MAX_FEATURES` (line 6). Line 27 then calls `calcOpticalFlowPyrLK()` and returns a number of areas in the `new_features` variable that have apparently shifted, compared with `ofeatures` in the last frame.

The `for` loop (lines 34-43) iterates across the areas and finds the range that covered the longest path. If one of them exceeds the value of 100, line 44 returns the value 1 from `move_test()`, thus indicating that movement must have occurred.

Spice It Up and Show Me!

Listing 1 thus outputs lines of integer values that represent the values in seconds for times at which something in the video

LISTING 3: motion-meta

```
01 #!/usr/local/bin/perl -w
02 use strict;
03 use File::Temp qw( tempdir );
04 use File::Copy qw( move );
05 use DateTime::Duration;
06 use DateTime::Format::Duration;
07 use Image::Magick;
08
09 my %seen = ();
10
11 my $video = shift @ARGV;
12 if( !defined $video ) {
13   die "usage: $0 video";
14 }
15
16 my $tmpdir = tempdir( CLEANUP => 1 );
17 my $magick = Image::Magick->new;
18
19 while( <> ) {
20   chomp;
21   my $second = $_;
22   next if $seen{ $second }++;
23
24   system "mplayer", "-ss", $second, $video,
25     "-vo", "jpeg:outdir=$tmpdir",
26     "-ao", "null", "-frames", 1;
27
28   my( $frame ) = glob "$tmpdir/0*";
29   my $newname = sprintf
30     "$tmpdir/frame-%s.jpg",
31       secs_format( $second );
32   move $frame, $newname;
33   $magick->Read( $newname );
34 }
35
36 my $montage = $magick->Montage(
37     label  => "%f",
38     shadow => "True",
39     tile   => "5",
40 );
41
42 $montage->Write( "motion-meta.jpg" ) and
43   die "write failed";
44
45 sub secs_format {
46   my( $secs ) = @_;
47
48   my $fmt =
49     DateTime::Format::Duration->new(
50       pattern => "%T" );
51
52   return $fmt->format_duration(
53     DateTime::Duration->new(
54       seconds => $secs )
55   );
56 }
```

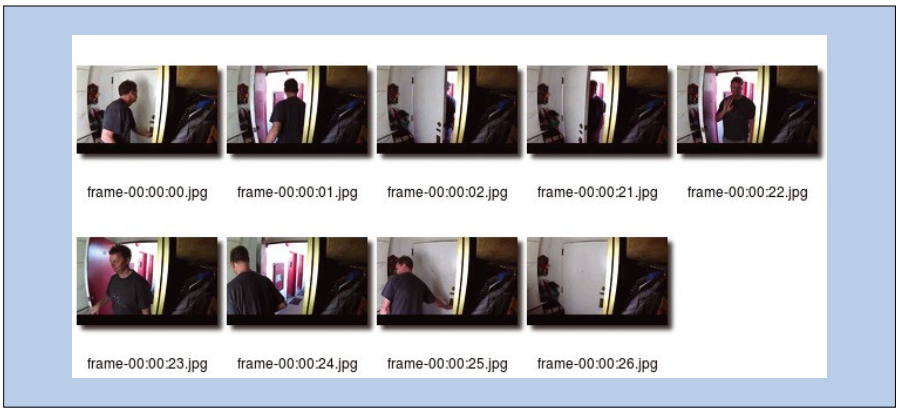


Figure 4: The motion filter only shows the seconds of video in which something actually happened.

moved from one frame to the next. It's now time for the code in Listing 3 to spice up this raw data, generate thumbnails for the appropriate scenes, and summarize the whole thing in an overview, as shown in Figure 4.

It uses the good old, all-around `mplayer` tool for the thumbnails, fast forwarding to the specified video second with the `-ss` option and storing the frame in a temporary directory `$tmpdir`. The `-frames 1` option stipulates that `mplayer` terminates right after reading a single frame. The `move()` function from the CPAN File::Copy module then renames the file in the temporary directory to one in the current directory and uses the CPAN DateTime::Format::Duration module to convert movie seconds to the `hh:mm:ss` format. The frame at second 64 thus becomes the file `00:01:04.jpg`.

The Ubuntu `perlmagick` package adds the CPAN Image::Magick module to the system, which you can use to create montages from multiple image files (i.e., virtual contact sheets) in the format shown in Figure 4. The call

```
$ max-movement-lk test.mp4 | ➥
  ./motion-meta test.mp4
```

glues the two parts of the pipeline together and produces the contact sheet in `motion-meta.jpg`.

The first part analyzes the frames in the video and prints the values (in seconds) during which movement has occurred. The second part grabs the movie second values, deduplicates them, looks up the associated thumbnail in the video, and mounts all of them to create a contact sheet – using nothing but the raw still image file names, conveniently chosen to reflect the time in the video in minutes and seconds.

10 Million for a Specialist

Detecting moving objects in video streams is used not only with surveillance videos, but with self-driving cars to distinguish endangered pedestrians from stationary road signs. Learning these techniques could pay off in terms of your career: According to ex-Googler Sebastian Thrun, companies in this field are currently outdoing each other and paying about $10 million for specialists [5]. Now who could afford it to say no to that? ■■■

INFO

[1] Arlo Security System: *https://www.amazon.com/dp/B00P7EVST6*

[2] Deniz Suarez, O., M. del Milagro Fernandez Carrobles, N. Vallez Enano, G. Bueno Garcia, and I. Serrano Gracia. *OpenCV Essentials*. Packt Publishing, 2014. *https://www.amazon.com/OpenCV-Essentials-Oscar-Suarez-2014-08-25/dp/B017YC0EHA*

[3] Lucas-Kanade method: *https://en.wikipedia.org/wiki/Lucas%E2%80%93Kanade_method*

[4] Listings for this article: *ftp://ftp.linux-magazine.com/pub/listings/magazine/195/*

[5] Post by Sebastian Thrun: *http://www.recode.net/2016/9/17/12943214/sebastian-thrun-self-driving-talent-pool*

Your Roadmap to the Open Hardware Revolution ...

An exciting world of projects, tips, and skill-building tutorials awaits you in every issue of Raspberry Pi Geek.

Order your subscription today and tune in to the revolution!

shop.linuxnewmedia.com

Print Sub

Carry our easy-to-read print edition in your briefcase or backpack – or keep it around the lab as a permanent reference!

Digital Sub

Our PDF edition is a convenient option for mobile readers.

6 print issues with 6 DVDs or
6 digital issues for only

$59.95 £37.50 €44.90

Download on the App Store

GET IT ON Google play

iTunes Store | Google US | Google UK

Security audits with Lynis

Auditor

Running a security audit periodically on your system lets you spot unexpected changes and possible weak points. *By Bruce Byfield*

Security is on everybody's mind these days, but where do you start? For that matter, how do you know when your precautions are complete? One answer to both of these questions is Lynis [1], which audits the security of a system by running more than 200 tests in a matter of a few minutes.

Lynis was created by Michael Boelen in 2007 and is now maintained by CISOfy [2], which uses Lynis as the back end for its commercial desktop application Lynis Enterprise [3]. It was inspired by Bastille Linux [4], which a decade ago was a standard Linux security tool but now is semi-moribund and no longer available in most distributions. The main difference between Bastille and Lynis is that Bastille included a system audit tool and a hardening wizard, which among other things could configure a firewall, whereas Lynis offers only an audit, leaving users to make changes for themselves. Another difference is that Bastille focused on specific releases of a few Linux distributions, whereas Lynis runs on most Unix-like systems without concern for the release number – a difference that is especially welcome with distributions that have rolling releases. In general, Lynis offers a more flexible and thorough audit than Bastille, reporting facts and leaving you to make the decisions.

Basic Syntax

Lynis is a shell script available as a package or tarball or as a GitHub clone [5]. Running it requires access to `/tmp`. For a complete system audit, Lynis should be run as root, but it can also be run from an ordinary user account for penetration testing. Documentation is available online [6] or from the man page, but it is written to include Lynis Enterprise, which means that not all the options listed are available in the free version. At times, the only way to know which options are available is to try them; fortunately, the unavailable

Lead Image © limbi007, 123RF.com

```
#
################################################################################

# Compressed uploads (set to zero when errors with uploading occur)
compressed-uploads=yes

# Show non-zero exit code when warnings are found
error-on-warnings=no

# Use Lynis in your own language (by default auto-detected)
language=

# Lynis Enterprise license key
license-key=

# Defines the role of the system (personal, workstation or server)
machine-role=server

# Profile name, will be used as title/description
profile-name=Default Audit Template

# Number of seconds to pause between every test (0 is no pause)
pause-between-tests=0

# Enable quick mode (no waiting for keypresses, same as --quick option)
quick=no
```

Figure 1: Lynis runs from a detailed, customizable profile.

```
root@nanday:/etc/lynis# lynis show tests
# Test       OS          Description
# ================================================================================
ACCT-2754  FreeBSD    Check for available FreeBSD accounting information (security)
ACCT-2760  OpenBSD    Check for available OpenBSD accounting information (security)
ACCT-9622  Linux      Check for available Linux accounting information (security)
ACCT-9626  Linux      Check for sysstat accounting data (security)
ACCT-9628  Linux      Check for auditd (security)
ACCT-9630  Linux      Check for auditd rules (security)
ACCT-9632  Linux      Check for auditd configuration file (security)
ACCT-9634  Linux      Check for auditd log file (security)
ACCT-9636  Linux      Check for Snoopy wrapper and logger (security)
```

Figure 2: Lynis runs more than 200 tests, including some specific to a distribution or operating system.

```
2016-12-05 15:35:17 Starting Lynis 2.4.0 with PID 17199, build date 2016-10-27
2016-12-05 15:35:17 ================================================================================
2016-12-05 15:35:17 ### 2007-2016, CISOfy - https://cisofy.com/lynis/ ###
2016-12-05 15:35:17 Checking permissions of /usr/share/lynis/include/profiles
2016-12-05 15:35:17 File permissions are OK
2016-12-05 15:35:17 Reading profile/configuration /etc/lynis/default.prf
2016-12-05 15:35:17 Action: created temporary file /tmp/lynis.qG862iMeeM
2016-12-05 15:35:17 Language set via profile to ''
2016-12-05 15:35:17 Plugin 'compliance' enabled according profile (/etc/lynis/default.prf)
2016-12-05 15:35:17 Plugin 'configuration' enabled according profile (/etc/lynis/default.prf)
2016-12-05 15:35:17 Plugin 'control-panels' enabled according profile (/etc/lynis/default.prf)
```

Figure 3: The Lynis logfile gives a thorough description of each audit.

options simply will not run and cannot harm your system.

Lynis runs from a profile located in `/etc/lynis/default.prf` that includes the tests to run (Figure 1). Other profiles for specific distributions like Debian or Red Hat Enterprise Linux will be run if Lynis detects that they are relevant. Users can also create their own profiles, although most will probably be content with the default.

Lynis's basic syntax is:

```
lynis AUDIT OPTIONS
```

The three major audit options are `audit system`, `audit system remote HOST`, and `system dockerfile FILE`. However, Lynis also has what the documentation calls "helper options," which are entered in the same position as the audit options. In particular, the `show helper` option displays information about the work directory and file locations. The tests (Figure 2) that Lynis runs are listed with the command:

```
lynis show tests
```

Most of Lynis' regular options affect how an audit is run. The option `--test TEST-ID` can limit the audit to specified tests, whereas `--pentest` runs a penetration test from a regular user account. For ease of reading, you might want to use `--wait`, which pauses between sections of the audit, giving you more time to read. You can also add `--quiet` (`-Q`) to run an audit without any user input. Other options change the default components: `--logfile PATH`, `--plugin-dir PATH`, and `--profile FILE` all being self-explanatory. For security reasons, you might prefer `--no-log` to prevent sensitive information from being written to disk.

Other options format the report that is output to the screen. With `--no-color`, the report uses only the foreground color set for the terminal, whereas `--reverse-colors` is useful against a light background.

If you do not add any options, or a circumstance arises not covered by the options added to the basic command, the output will add brief help notices as needed.

Running an Audit

As you gain experience with Lynis, you might want to experiment with some of its options. However, you can make quick, practical use of Lynis with the bare command:

```
lynis audit system
```

The audit outputs to the screen, writing the information to `/var/log/lynis-report.dat`. To get a more exhaustive view of the audit, view `/var/log/lynis.log` (Figure 3). Both the report data and the logfile are overwritten when the next audit is run, so you need to rename these files if you regularly audit the system. You might also copy and paste the report directly from the screen into a file.

So far as possible, the report remains neutral. It reports whether useful resources are available or not and offers detailed warnings only at the end. So far as I can see, it recognizes Systemd as a service manager but does not check for its security particulars. Nor does it mention Firejail [7], which provides a measure of security by containerizing standard applications. The audit takes a classic architectural stance, focusing on the security built in to the system rather than features like antivirus software.

The report begins with general information about the system and the Lynis settings used and checks for key files and configurations, including those specific to Debian or another popular distribution. These resources are reported variously as FOUND, NOT INSTALLED, DISABLED, NONE, NON-DEFAULT, DIFFERENT, NOT RUNNING, or WARNING (Figure 4). Ordinarily, only WARNING is an immediate concern, since a file or setting may be not installed or disabled without necessarily being a security risk. Possibly, too, where there is a choice, as with the boot manager, only one is installed. Still, you might want to check these results, just to be sure, after you deal with the more urgent results.

```
[+] Boot and services
------------------------------------
  - Service Manager                                           [ systemd ]
  - Checking UEFI boot                                        [ DISABLED ]
  - Checking presence GRUB2                                   [ FOUND ]
    - Checking for password protection                        [ WARNING ]
  - Check running services (systemctl)                        [ DONE ]
        Result: found 32 running services
  - Check enabled services at boot (systemctl)                [ DONE ]
        Result: found 26 enabled services
  - Check startup files (permissions)                         [ OK ]

[+] Kernel
------------------------------------
  - Checking default run level                                [ RUNLEVEL 5 ]
  - Checking CPU support (NX/PAE)
    CPU support: PAE and/or NoeXecute supported               [ FOUND ]
  - Checking kernel version and release                       [ DONE ]
  - Checking kernel type                                      [ DONE ]
  - Checking loaded kernel modules                            [ DONE ]
      Found 116 active modules
  - Checking Linux kernel configuration file                  [ FOUND ]
  - Checking default I/O kernel scheduler                     [ FOUND ]
  - Checking for available kernel update                      [ OK ]
  - Checking core dumps configuration                         [ DISABLED ]
```

Figure 4: Lynis reports on a system's resources in depth with various summaries.

```
-[ Lynis 2.4.0 Results ]-

Warnings (2):
----------------------------
! Found one or more vulnerable packages. [PKGS-7392]
    https://cisofy.com/controls/PKGS-7392/

! Couldn't find 2 responsive nameservers [NETW-2705]
    https://cisofy.com/controls/NETW-2705/

Suggestions (38):
----------------------------
* Install libpam-tmpdir to set $TMP and $TMPDIR for PAM sessions [CUST-0280]
    https://your-domain.example.org/controls/CUST-0280/

* Install libpam-usb to enable multi-factor authentication for PAM sessions [CUST-0285]
    https://your-domain.example.org/controls/CUST-0285/

* Install apt-listbugs to display a list of critical bugs prior to each APT installation. [CUST-0810]
    https://your-domain.example.org/controls/CUST-0810/
```

Figure 5: The suggestions and warnings in the report link to web pages with more detailed information.

For example, the maximum age for a password before it must be replaced is usually disabled in most major distributions without major consequences. However, if you are security conscious or the system is networked, enabling this feature adds security with minimal effort. Similarly, not having a default `umask` – the default permissions for a new file or directory – might seem minor, but it is a bit of hardening generally worth implementing. By contrast, non-default `mount` options for your partitions probably means nothing more than that you have customized them; in fact, altering them will probably reduce overall performance. In the same way, if a service like CUPS is not running, then you need to turn it on and rerun the audit so that it can be accessed.

The report finishes by summarizing warnings and suggestions (Figure 5). The warnings will include both lax configurations and vulnerable packages, whereas the suggestions spell out a concrete action that either you need to do only once or recommend the addition of a tool that will help with ongoing security. Both warnings and suggestions are accompanied by a link to more detailed information on a CISOfy page (Figure 6), worded in such a way as to help you decide whether you should implement it or not. However, in a few cases, you might want to search for more information; although this can be a tedious process, it at least organizes and makes sense of the complicated subject matter and reduces much of the uncertainty.

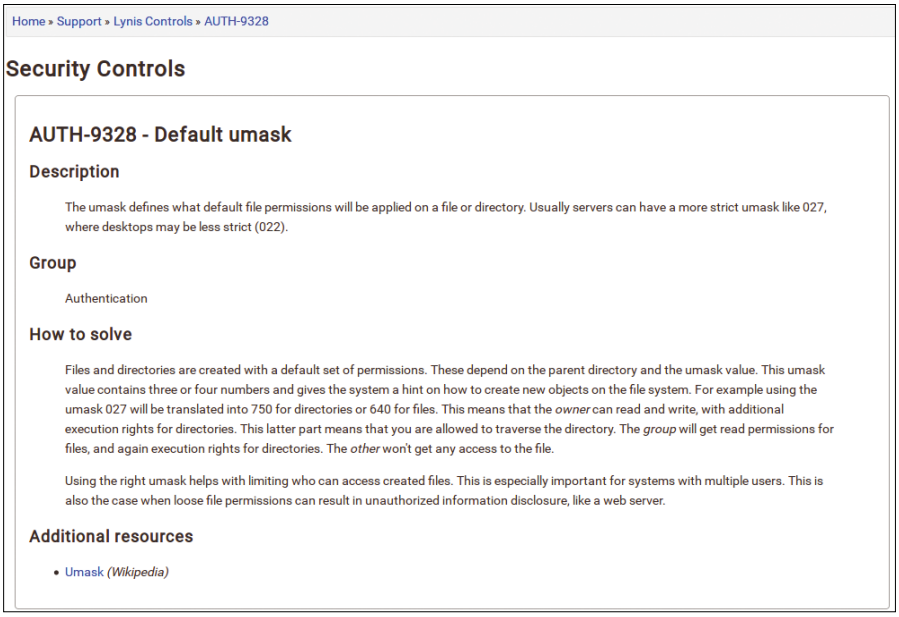


Figure 6: The final payoff: Concrete suggestions about how to improve the system.

After the First Report

Lynis should be run as soon as it is installed and then acted on as necessary. It should also be run again after you have acted on the warnings and suggestions. However, that is just the start. Running Lynis regularly is a convenient way to spot unexpected changes in a system that might indicate intrusions. Moreover, your needs might change over time, and Lynis' report can help you decide what adjustments are necessary.

Those who once depended on Bastille are likely to find Lynis a less complete solution. If nothing else, Bastille did a more thorough job of explaining the pros and cons of possible actions. Yet, despite this shortcoming, Lynis remains a quick and convenient update of the Bastille concept and an education in itself. So long as you are willing to put in the time, you can benefit without being a security expert. If you have never run Lynis on a system that you administer, you owe yourself the favor of doing so immediately. ■■■

INFO

[1] Lynis: *https://cisofy.com/lynis/*
[2] CISOfy: *https://cisofy.com/*
[3] Lynis Enterprise pricing: *https://cisofy.com/pricing/*
[4] Bastille: *http://bastille-linux.sourceforge.net/*
[5] Lynis on GitHub: *https://github.com/CISOfy/Lynis*
[6] Documentation: *https://cisofy.com/documentation/lynis/*
[7] Firejail: *https://firejail.wordpress.com/*

Ben Everard

It's no secret that 2016 has brought a lot of change to the world. The political destinies of both Europe and America have shifted in ways that seemed almost inconceivable just a few years ago, and that's without mentioning the scores of beloved celebrities that are no longer with us. Here in the UK there's been yet another political shift this month, though one we've long seen coming. We've followed the progress of the Investigatory Powers Bill (also known as the Snooper's Charter) with dismay since it was first proposed just over a year ago. On November 16, 2016, the act passed through the final chamber of parliament and only waits for the Queen's signature before becoming law. The UK is now poised to invade privacy by digitally tracking every citizen, and the government has the right to hack into our devices. Digital security has always been important, but now it's more so than ever. This month Valentine Sinitsyn looks at auditd, and I take a look at intrusion detection.

Even our cozy Linux world hasn't escaped the perversions of 2016. November also brought the shocking news that Microsoft joined the Linux Foundation. It's a little hard to process information as absurd as this, so we've harnessed the finest minds we could find (Simon Phipps and Andrew Gregory) to bring a little sense to this madness.

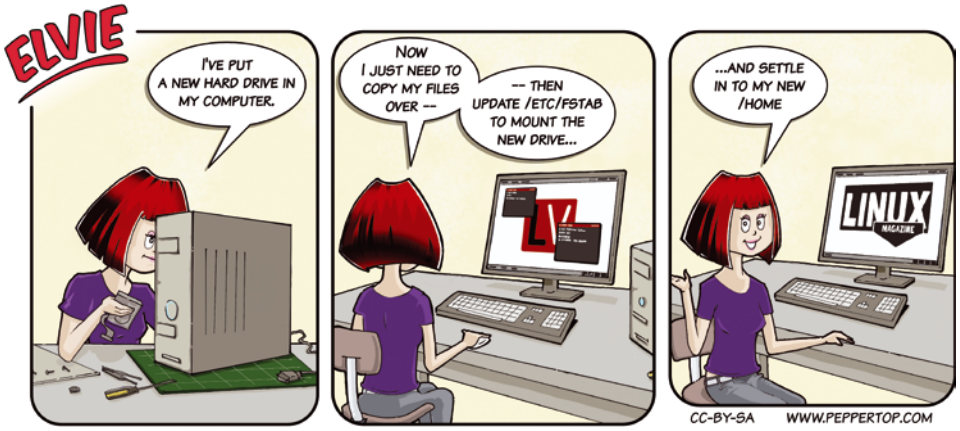
Not everything has changed (not yet, at least, but 2016's not over), and there are still few better ways to survive the cold, dark winter months than settling down with a good game, immersing yourself in a different reality. In Gaming On Linux this month, we take a look at the latest triple-A masterpiece to come to our platform – Deus Ex: Mankind Divided. Alongside all this, there's plenty more Linux goodness to help banish the 2016 blues, so turn the page and get stuck in.

– Ben Everard

Andrew Gregory

Graham Morrison

Mike Saunders



LINUXVOICE

NEWSANALYSIS

The Linux Voice view on what's going on in the world of Free Software.

Opinion

Has Microsoft Surrendered to Open Source?

Hey, everybody, they are still Microsoft. BY SIMON PHIPPS

Simon Phipps is ex-president of the Open Source Initiative and a board member of the Open Rights Group and of The Document Foundation (makers of LibreOffice).

One of the big stories of 2016 has been Microsoft's continuing charm offensive to persuade the open source free software communities they are now "us" rather than "them." They've changed: joined Eclipse, publicized their contributions to large numbers of projects, and even joined The Linux Foundation at the highest (Platinum) level.

In response, almost every commentator has been lauding Microsoft's progress and dismissing residual criticisms – almost certainly the response Microsoft seeks. One veteran commentator even declared "open source has won and Microsoft has surrendered" [1].

To believe that Microsoft has "surrendered" is not right. They are still a company embracing the theory of maximizing shareholder value regardless of morality. They continue to do so without regard for the social norms of the communities they find in their chosen markets. That's demonstrated in part by their continuing monetization of their software patent portfolio at the expense of the other members of the Linux Foundation, which they have recently joined.

As an indication of the scale of that monetization, Linux Foundation's $500k Platinum membership fee [2] represents about one quarter percent of the lowest estimate [3] of Microsoft's income from asserting patent threats against the Linux kernel and things that use it such as file servers, Chrome OS, and Android. That lowest estimate – $2bn – represents the total revenue of genuine open source Linux leader Red Hat. At around 2% of Microsoft's revenue, it's a considerable sum, but still one that could feasibly be foregone in support of business strategy, given the will.

So it's not Microsoft's whole business that's changed to favor open source. Office is not available for Linux, and neither are Sharepoint, Active Directory Server, or indeed any other paid software product. What *has* changed is they have dropped the legacy leadership of their founders that saw it as impossible to embrace open source in the pursuit of shareholder value. Those legacy leaders could not tolerate the presence of copyleft software within any strategy, for what seemed a lot like ideological reasons from the outside.

The key change is thus new leadership that can simultaneously visualize proprietary monetization of "boxed products" and a belief in the relevance of copyleft community software. The result is they now have a division with executive aircover – their Azure cloud product division – that understands its future success depends on hosting Linux as a runtime environment and appealing to the developers who target it.

This business unit is an underdog in its market and has a strong motivation to embrace open source to progress. That business unit is doing a great job, and it is exerting influence on other parts of the company to play along. The Azure team deserves encouragement and praise for its stance toward open source. They are very effectively pursuing both hosters and developers and making friends in the open source community.

I seriously doubt they have embraced the commitment to developer and user freedom from which open source flows. They are maintaining proprietary control over Azure, as well as integrating Linux run-time capabilities into Windows. Whereas Red Hat has a corporate commitment to open source that results in their new products trending to open source, Microsoft's instincts trend in the opposite direction. They'd like to make Linux servers merely Azure instances and make the desktop into Linux for Windows services under Windows, rather than offering software freedom. If left unchallenged, that's the future of mainstream Linux – APIs in a proprietary container.

One can either speak just of Microsoft's cloud business and its related developer community and credit them with open source goodness, or one can speak of Microsoft and balance the actions of the various parts of the company with radically different attitudes. We want Microsoft in the community, but not at the cost of turning a blind eye to patent parasitism, to the containment of software freedom in a proprietary casing, and to the ostracism of community members who dare to find fault in either.

No, Microsoft has not surrendered, except in the sense of Greene's 22nd Law [4]. They simply have a leadership that can see open source is of primary importance in securing a leadership share in utility cloud computing. The company has indeed changed, but that change has not resulted in embracing the origins of software freedom, just the fruits of one part of its company. ■■■

Info

[1] S.J. Vaughan-Nichols opinion piece: *http://www.computerworld.com/article/3144063/open-source-tools/open-source-has-won-and-microsoft-has-surrendered.html*

[2] LF bylaws: *https://www.linuxfoundation.org/about/bylaws*

[3] Microsoft's Android royalties: *http://www.businessinsider.com/microsoft-earns-2-billion-per-year-from-android-patent-royalties-2013-11*

[4] Greene, R. *The 48 Laws of Power*: *http://48-laws-of-power.blogspot.co.uk/2009/01/law-22-use-surrender-tactic-transform.html*

Linux Foundation eunt domus!

BY ANDREW GREGORY

We're gonna need a bigger bazaar.

Hell has frozen over, the Eagles have re-formed, and Microsoft has joined the Linux Foundation. We must conclude that the Linux Foundation has zero credibility, so now is the time for the Free Software Foundation (FSF) to step up and provide not just the moral leadership for Free Software, but some sort of authentic advocacy.

Ha! Had you going for a bit there, didn't I? While the Linux Foundation is sullied beyond redemption, the FSF cannot become the campaigning, pro-Linux, pro-business voice while it prizes ideological purity above all else. To be clear, the FSF is magnetic north in terms of purity. It's immovable and incorruptible, and when it speaks, we should listen; however, it doesn't do a good job of promoting Free Software. We need a compass to point the way, and that cannot any longer be the Linux Foundation.

So in the spirit of the splitter, I propose a new organization: the People's Popular Front of Linux (PPFL). This organization would campaign tirelessly to raise public awareness of Linux on the desktop, on the smartphone, and on the server. It would be ready with a handy quote for journalists whenever there's a story about "Computer Viruses," informing the public that it's actually Microsoft that's at fault rather than computers in general. It would be there to provide insight the next time closed-source code is implicated in a car emissions cheating scandal, or in a hacking incident, or when someone's closed-source insulin pump goes wrong because of a hard-coded error and the license makes it illegal for anyone other than the vendor to fix it.

When a self-driving car makes you late for your appointment because it needed to re-boot and install an update, the PPFL will be there to tell the world that it's not acceptable; the Linux Foundation will shrug its shoulders and count the money. When an ISP stores its passwords unencrypted and literally everybody in the mainstream ignores the cavalier risks they took because everyone else is taking them, the PPFL will speak up and advise on best practice. When the Internet of Things finally reaches into every corner of our lives, the PPFL will need to shout from the rooftops that if you don't control the software, the software will end up controlling you.

Plenty of experts are already working to protect the freedoms that we gain as software users, including the OSI, the Software Freedom Law Center, the Open Rights Group, and the Electronic Frontier Foundation, but when the holder of the Linux trademark is in the pocket of Microsoft – an organization with a natural opposition to Free Software, Linux, and open source – it's just not fit to represent Linux any more. We need a better alternative and we need it now. Who's with me? ■■■

Jon "maddog" Hall is an author, educator, computer scientist, and free software pioneer who has been a passionate advocate for Linux since 1994 when he first met Linus Torvalds and facilitated the port of Linux to a 64-bit system. He serves as president of Linux International®.

MADDOG'S DOGHOUSE

No matter who the US President is or why, it's the citizens who need to understand how they can prepare for the jobs that are begging to be filled in modern economies. BY JON "MADDOG" HALL

On November 8, 2016, the United States was rocked by a historic election. Donald Trump was elected the 45th President on a campaign slogan of "Make America Great Again."

Whether or not you agree that the United States is or ever was great, or even that the word "America" is misused when people talk about the United States, his election by a large portion of people who feel that the economy has left them behind is something that (in retrospect) most of the "professional politicians" have ignored. Whether Donald Trump and his policies will make any difference is to be seen in the future; however, this issue is not just confined to the United States, it is happening all over the world.

In the short term, I do not think governments can do very much to keep jobs in-country, because the problem is not just about the transference of jobs to another part of the world. Certainly that is what most people see, and it lends to the problem, but the real issue is that the kinds of jobs that new and evolving companies offer do not lend themselves, because of automation and mechanization, to the kind of education that most people today receive – or have received in the last 10 years.

Many people today recognize that there is a lack of prospective employees with a STEM (science, technology, engineering, and math) education. At the same time that unemployment was reaching its heights, many companies were complaining that they could not find the trained people they needed for their businesses. Unfortunately, STEM-educated people cannot be produced overnight. It requires a solid education in math, practical physics, and general chemistry and biology, most of which are lacking in students who graduated from high schools and universities in the last 10 or 20 years. Please do not think that this is an attack on our educational professionals. If anything, this article is a wake-up call to parents and students of all ages everywhere – not just in the United States.

Competition (and this is no attack against capitalism, either) is going to keep driving the automation of various jobs, which will cause the STEM education needs to go ever higher. Even if a job will not be fully automated, the tools employees use to do the job consequently reduce the number of people needed.

I watched as CAD tools reduced the number of draftsmen (a highly skilled profession) in a large company from 300 to three. Draftsmen no longer had to redraw entire large sheets of paper to make simple changes. A few quick commands to a program would copy the entire design and allow them to make a quick update. This was not a company moving their work to China or some other country. The jobs just went away.

As a computer engineer, I do not regret this path, but I do see the problem getting even more focused in the future, which is why the students of today (even students that are 40 years old) have to face the fact that it will be up to them to get the education and training that they need and to continue being trained for the rest of their lives.

Which brings me to Free and Open Source Software and Hardware.

In the past, automobiles were mostly mechanical. Most people with rudimentary skills in mechanics could understand how an automobile worked and repair it. Young people could work on old cars – "tinkering" with them – and learn how to repair newer cars.

Today, more and more of the mechanical parts are replaced by a simple sensor and a microprocessor, which is typically "closed source." Young "auto mechanics" cannot look at the source code of older cars and see how a newer car works. Cars (and the software that drives them) keep getting more and more sophisticated. Where will we get the auto mechanics of the future, the people trained and knowledgeable about this software?

The same could be said about operating systems, compilers, and the software that drives the cloud. We need more open systems to allow the tinkerers to learn and grow. Five years ago the Raspberry Pi Foundation created a sensation with their small computer that was designed (for the most part) by university professors who realized that new freshmen were not as knowledgeable about computers as the students of 20 years ago. By providing an open platform for high school students to use, the professors made their teaching jobs easier by having students arrive at the university with a higher level of computer knowledge.

For the level of STEM needed in the world today, grades nine through 12 are too late to introduce STEM education. Educators, parents, and the students themselves need to stimulate interest and thirst for STEM knowledge from birth and throughout their lives.

A knowledgable employee is not something that one man or political party can create by themselves; to paraphrase, it takes a village to make this happen. ■■■

Now Appearing on
APPLE NEWSSTAND

New age convenience...

Our inspired IT insights are only a tap away.

Look for us on Apple Newsstand and the iTunes store.

Download a FREE issue of each publication now!

GoboLinux: Out with the old, in with the new

Explore this novel Linux distro, which throws out the old Unix filesystem hierarchy in favor of something more modern.

BY MIKE SAUNDERS

One of my favorite sayings comes from the mouth of Henry Spencer, author of the Regex regular expression parsing library (among many other things). He said: "Those who do not understand Unix are condemned to reinvent it, poorly." But, what exactly did he mean by this? Well, many hackers regard Unix as the pinnacle of operating system design. Unix pieces together many ideas and technologies that make it a truly universal operating system, running on everything from wristwatches to supercomputers. The concepts and foundations on which it's based will be solid forever.

Still, every few years, someone comes along promising a revolutionary OS that discards all the old 1970s baggage and implements everything using the latest buzzwords du jour. Why implement all that old Unix gunk when you can build an OS in a version 0.0.3 language someone just posted on Hacker News? Why should everything be a file when obviously JSON is the only sensible way to store data? And who needs a filesystem anyway?

But the same thing tends to happen to trendy OSes like these. Over time, they either die off or end up becoming more and more Unix-like. The developers realize that Unix isn't actually old and clunky but is actually a rather smart design crafted by hackers with a lot of experience. Unix may have been written for the mainframes and minicomputers of the 1970s, but it was very easily adaptable to smaller devices. It's no coincidence that Unix flavors pretty much run the world now: Linux and FreeBSD on servers, Android on phones, Mac OS on many desktops and laptops. Even Microsoft is trying to get in on the action with its Windows Subsystem for Linux.

So Unix is a solid and well thought-out OS, and straying too far from its principles can be risky. However, that doesn't mean you should be overly dogmatic and resist any change. Arguably, a few things in Unix-like systems are a bit archaic and could be updated, without radically re-engineering the whole OS. Say hello to GoboLinux [1].

GoboLinux is – for the most part – a regular Linux distribution, but with one major difference: it eschews the old filesystem hierarchy (`/usr`, `/lib`, `/etc`, and so forth) in favor of a more modern design where each program lives in its own directory. In theory, this should make it easier to distribute, install, and remove programs. Additionally, GoboLinux features its own boot system that's not based on the old SysVinit or BSD scripts – nor does it use Systemd. It's a fascinating distro, so in this article, I'll show you how it works, how to install it, and where it's heading.

A Bit of Background

Most Linux distributions adhere to the Filesystem Hierarchy Standard, aka FHS [2], which determines the directory structure in a typical Linux installation – i.e., which files and directories go where on the drive. The FHS is heavily inspired by older Unix flavors and leads to a setup like this:

/ (aka root directory) – The highest level directory, with nothing above. This contains all other directories (and their subdirectories).

/etc – Configuration files. This really is a mixed bunch, containing text-based config files for system-wide applications and services, scripts, and other bits 'n' pieces. There's very little standardization in here, with each file having its own format and syntax, and various subdirectories.

/bin and /sbin – Base-system binaries for regular users and the superuser (root), respectively. These are available immediately after booting, before other filesystem partitions are mounted. Here's where you find the Bash shell and other tools.

/lib – Historically, this contained base-system libraries, like the C library that almost every program depends on. Today, it also contains Linux kernel modules, systemd scripts (on distros that use it), and various other components.

/usr and /usr/lib – Programs and libraries for normal user accounts. This is typically the largest part of an installation and includes non-base software – the X Window System, desktop environments, Firefox, etc. In many older Unix installations, `/usr` would be on a separate partition and mounted later during the boot process to keep user apps separate from the base system.

Those are some of the most important directories in a typical Linux installation, but there are many more as well. You can see some historical baggage in there, with some directories serving multiple purposes, bits of overlap, and not much in the way of standardization (despite the best efforts of the FHS). Most significantly, though, is that a single program may have files in many different directories. Have a look at some packages on your current Linux installation and see where they scatter files. For instance, on a `.deb`-based distro:

```
dpkg -L bash
```

This shows that the Bash package drops files into `/etc`, `/bin`, `/usr`, and various sub-directories of `/usr`. There is some logic to this, but it makes a package manager pretty much essential. How can you easily remove a program when its files are scattered over so many different directories? How can you tell which file belongs to which program? Tools like dpkg, apt-get, yum, rpm, etc. take the pain of this away, but you could argue that they are just more layers of complexity and abstraction on top of what should be a simple design.

How does GoboLinux approach this? Essentially, it tries to redefine the filesystem. Instead of programs being scattered all over the drive and hard to manage without specific tools, each program is placed in its own subdirectory of `/Programs`, so they are easy to locate, change, and remove. In GoboLinux, "the filesystem is the package manager," drastically simplifying the software management process. On top of that, the distro makes it possible to have multiple versions of the same program installed, side by side. Try doing that with your regular Linux distro – even with the most awesome command-line gymnastics and symbolic link shenanigans, it's tough to achieve.

Installing GoboLinux

You may be reticent at this stage to switch to GoboLinux as your primary distro. Fair enough – I recommend exploring it in a virtual machine such as VirtualBox (or just installing it on a spare PC). Go to the download page [3] and grab the ISO image of the latest release – 016 beta at the time of writing, but it may be final when you read this.

In VirtualBox, assign it to the emulated DVD drive and start the virtual machine. After a few moments of booting up, GoboLinux will ask you to select a language and keyboard map. Then you'll land on a root command-line prompt – which probably wasn't what you're expecting, so enter `startx` to bring up the X Window System and access the GUI. From there, you'll find an installer icon on the desktop, so double-click that and follow the prompts.

If you have trouble running `startx`, though, you'll need to perform a text-mode installation (Figure 1) – enter:

```
Installer
```

Note the capital letter at the start – this is your first taste of how GoboLinux does things! Use Tab to switch between buttons and other elements on the screen, and Enter to select them. You'll be prompted to choose the partition(s) on your drive where GoboLinux should be installed, the package sets to be copied over (it's best to leave them all selected), and where to install the boot loader. For the most part, you can just accept the defaults and keep selecting *Next*.

You'll be prompted to set a password for root, the administrator user, and also create a normal user account for your day-to-day work. Lastly, the GoboLinux installer will perform a final check that you're ready to go ahead with the options you selected – if so, the files will be installed and then you can reboot. If GoboLinux boots into text mode, you can log in as your regular user account and enter `startx` to bring up the desktop (Figure 2).

Filesystem Magic

To see how GoboLinux handles software, open a terminal and `cd` into the `/Programs` directory. There you'll see a bunch of directories, one per program, so try switching into one – e.g., Nano. Then you'll see three directories: one with a version number, which contains the binary and support files for that version of Nano. There's also `Current`, which is a symbolic link to the version currently in use (this is how you can have

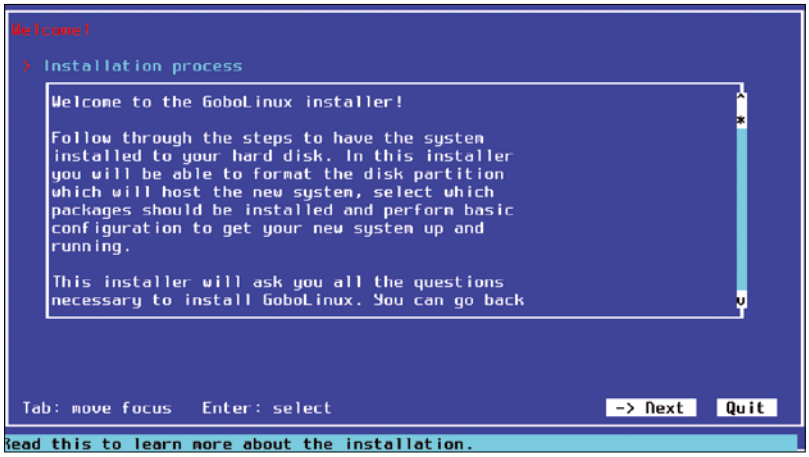


Figure 1: GoboLinux's text-mode installer gets you set up easily – you can mostly just hit Enter.

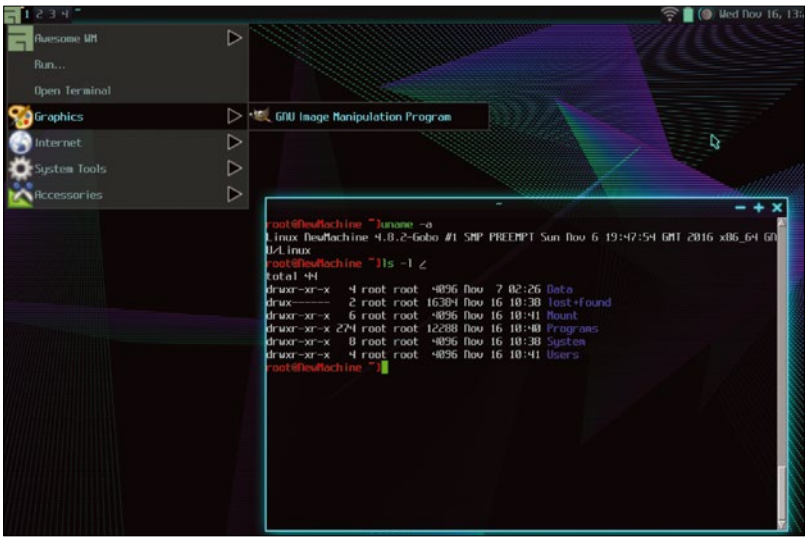


Figure 2: The Awesome window manager is included with version 016, along with a rather moody theme.

multiple versions installed at any one time), and `Settings`, which contains files that would normally be stored for system-wide usage in `/etc` (e.g., nanorc).

This system doesn't just apply to programs but to libraries as well. For instance, Glibc (the GNU C Library) can be found in `/Programs/Glibc/2.24/lib`). As for "infrastructure" software – i.e., not standalone desktop apps – those are in `/Programs`, too. You'll find the X Window System in `/Programs/Xorg-server`, binaries from Coreutils in `/Programs/Coreutils/Current/bin`, and so on.

This is all good and well, but how does GoboLinux keep track of all these, without having to use a giant `$PATH` environment variable? Well, take a look in the `/System/Index` directory. There you'll see `bin` and `lib` directories containing huge amounts of symbolic links to all of the programs and libraries stored separately in the `/Programs` folder. For instance, go into `bin` and do `ls -l nano`, and you'll see something like:

```
nano -> ../../../Programs/Nano/2.7.0/bin/nano
```

Figure 3: Here you can see what the root directory contains – /etc, /lib, and so on are all hidden by default.

```
root@NewMachine ~]cd
root@NewMachine /]ls -la
total 52
drwxr-xr-x  12 root root  4096 Nov 16 13:50 .
drwxr-xr-x  12 root root  4096 Nov 16 13:50 ..
drwxr-xr-x   4 root root  4096 Nov  7 02:26 Data
drwx------   2 root root 16384 Nov 16 10:38 lost+found
drwxr-xr-x   6 root root  4096 Nov 16 10:41 Mount
drwxr-xr-x 277 root root 12288 Nov 16 13:50 Programs
drwxr-xr-x   8 root root  4096 Nov 16 10:38 System
drwxr-xr-x   4 root root  4096 Nov 16 10:41 Users
root@NewMachine /]mount
/dev/sda1 on / type ext4 (rw,relatime,data=ordered)
devtmpfs on /dev type devtmpfs (rw,relatime,size=1534072k,nr_inodes=383518,node=
755)
proc on /proc type proc (rw,relatime)
none on /sys type sysfs (rw,relatime)
devpts on /dev/pts type devpts (rw,relatime,mode=600,ptmxmode=000)
none on /System/Kernel/Objects type sysfs (rw,relatime)
proc on /System/Kernel/Status type proc (rw,relatime)
devtmpfs on /System/Kernel/Devices type devtmpfs (rw,relatime,size=1534072k,nr_i
nodes=383518,mode=755)
devpts on /System/Kernel/Devices/pts type devpts (rw,relatime,mode=600,ptmxmode=
000)
```

This shows you how GoboLinux knows where to find Nano (and other programs) in the installation. The `/System` directory contains other components, neatly arranged into their own sub-folders, as well. You'll see that the kernel can be found there (along with its modules), while the `Settings` directory does the job of a traditional `/etc`.

Now, if you `cd` into the root (/) directory and look around with `ls -a`, you'll notice something odd: Apart from `lost+found`, none of the usual Unix-y filesystem locations can be seen. There's no `/etc`, `/lib`, `/tmp`, or anything like that (Figure 3). What's going on here? Has GoboLinux done away with them all completely?

Well, not quite. GoboLinux is set up with a kernel patch that hides these directories by default. They do actually exist, so you can `cd /etc` and then look around inside, but they're not shown to you in your daily work. This is all about simplicity: Users and admins shouldn't need to be concerned about the "legacy" Unix filesystem, apart from in extreme cases. For most tasks of editing config files, updating programs, and the like, these should all be possible via the newer GoboLinux filesystem structure.

Adding New Software

GoboLinux is primarily a source-based distro, so whenever you want a new piece of software, you issue a command that downloads the program's source code (along with the code for its dependencies) and builds the whole caboodle. First, take a look at the available "recipes" [4] – that is, the scripts and information bundles for various open source apps and tools (Figure 4).

Let's say you want to install Qemu, the rather awesome PC (and other machine) emulator. Enter the following command:

```
Compile qemu
```

This will search for a matching recipe and tell you what it's going to download. If the app has a lot of dependencies, `Compile` will prompt you before grabbing them; enter "CA" for "compile all" to skip having to confirm for every single file.

Then, `Compile` builds the dependencies and the program itself and installs them into GoboLinux creating symbolic links accordingly so that older programs using Unix-like paths can still work with them. GoboLinux has a lot of recipes at its disposal, so you can find pretty much every major FOSS app and utility in there, although some of the versions are rather dated (Figure 5).

Helpfully, though, the distro includes a small tool for changing this behavior. Enter `gobohide -l` to list all of the Unix-ish directories that are hidden by default. To disable hiding, so that one is shown at the command prompt and in your file manager, use the following:

```
gobohide -u /etc
```

Now `/etc` will be visible. You'll see that it's a symbolic link to the `/System/Settings` directory you were in earlier, however, so it's not just duplicated content.

Have fun exploring the system: Clearly a lot of thought has been put into the design, and it works surprisingly well given the amount of software out there that expects traditional Unix-style filesystem layouts. GoboLinux feels well engineered, and when you go back to a regular distro with its package files scattered all over the place, everything seems rather disjointed and thrown together.

Still, GoboLinux has its rough edges: I had issues getting some apps to work (see the "Adding New Software" box), and there are some questions about its long-term viability (there was a major gap of six years between releases 014 and 015). But even if GoboLinux isn't going to be your daily driver any time soon, it's good to try out a distro that dares to do things differently – after all, as great as Unix and Linux are, you should always give room for improvements. And, given that GoboLinux's novel directory structure can still be used to create a working distro, I'd like to see some elements of it incorporated into the mainstream distros at some point (see the "NixOS: Another Alternative" box). Fedora devs, what do you reckon? ■■■

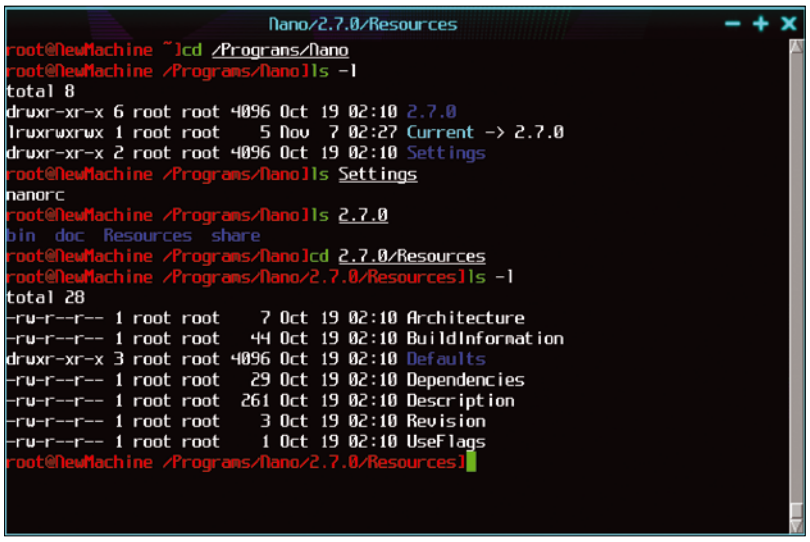


Figure 4: Look into /Programs/Nano and you can see how GoboLinux handles multiple app versions and meta information.

NixOS: Another Alternative

If GoboLinux has whetted your appetite for Linux distros that take an alternative approach to package management, take a look at NixOS [5] as well. This is built around Nix, a "fully functional package manager for Linux and other Unix systems that makes package management reliable and reproducible." The reproducible part is built upon checksums and other techniques that guarantee that binary packages match their source code equivalents. (In other words, nothing dodgy has been injected by the package creator in the building process.)

In terms of reliability, Nix packages exist in isolated read-only directories; updates don't overwrite the original files, so it's easy to revert back to the previous version of an app or library if you're having trouble with the new one. And, all updates are "atomic": They either succeed or they don't. So, if you're updating an app and the process gets interrupted half way (e.g., due to power loss), you won't end up with an app where half the files are from one version, and half from another.

While Nix has been quite innovative here, some other mainstream distros are starting to adopt similar approaches. Take Ubuntu, for example, with its Snap packages [6] – they are also designed to be updated safely and are isolated from one another so that one badly behaving app can't cause chaos on the whole system.

Info

[1] GoboLinux: *http://www.gobolinux.org*

[2] Filesystem Hierarchy Standard: *https://wiki.linuxfoundation.org/lsb/fhs*

[3] GoboLinux download: *http://www.gobolinux.org/?page=downloads*

[4] GoboLinux recipes: *http://recipes.gobolinux.org*

[5] NixOS: *http://nixos.org*

[6] Snap packages: *http://snapcraft.io*

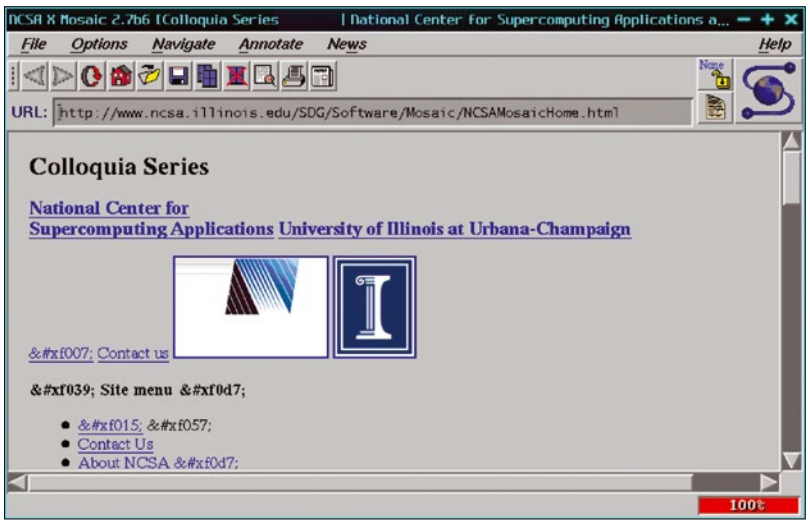


Figure 5: For some reason, GoboLinux includes the ridiculously outdated Mosaic browser. It loads almost nothing.

Linux's Next-Generation Firewall

Nftables promises to be the future of Linux firewalls. Meet iptables' replacement.

BY BEN EVERARD

Q Why should I be concerned about what Christian hip-hop artist Nate Feuerstein does with his tables?

A Not NF's tables, but nftables (or Netfilter tables). It's the name of the next-generation firewall for Linux.

Q Firewall? That sounds like a pretty horrible thing. Even if the fire doesn't spread, the smoke damage would be catastrophic for the interior decoration. I can't imagine it'd be too good for the computer either, especially as the fan that's supposed to keep the CPU cool would only make the fire hotter. You can keep your pyromania to yourself, thank you very much.

A Just on the off chance that you're not being deliberately difficult, a firewall is a bit of software that looks at all the network traffic going in or out of your machine and decides if it should be let through or not. These can be as simple or complex as you like. For example, a really basic firewall would let any outbound connections through (so you could get data from other machines) but not any inbound connections (so no one could request data from you). If you then decided to run a web server on this machine, you'd have to open up port 80 so that browsers could connect to you.

Q Hang on, I'd need 80 ports? I've only got one Ethernet connection. Where would I get the other 79?

A In IP networking (which is almost all networking these days), you need two identifying things when connecting to the other computer. There's the IP address, which locates the machine to which you want to connect, and there's a port number, which identifies the software you want to connect to on that machine. The port number doesn't refer to a physical port but is just a reference, so the machine receiving the connection knows where to send the data.

On any machine, only one piece of software can be listening on a given port, so the kernel forwards all data to that software. All ports are the same, so in principle you could have any software listening on any port number, but a few conventions make life a little easier. For example, web servers usually listen on port 80 (for HTTP) and 443 (for HTTPS), so when you use a web browser, it will always try to use these ports unless you specify a different port, which is done by adding a colon and then the port number after the web server domain name.

Q Ah, OK. So all I have to do is tell my firewall what services I want to run, and it'll open up those ports?

A Well, that would be a very basic setup. Firewalls can do quite a lot more than this, though. The current standard Linux firewall, iptables, allows you to do things like limit access to certain services to particular machines (usually identified by IP address), log particular events, and even block people who are repeatedly trying to guess passwords. All together, the iptables firewall is a large part of most Linux setups' security.

Q This all sounds quite useful. I should get a firewall.

A If you're using Linux, you probably already have one, and it's probably iptables. You can see what it's currently doing by entering the following at the command line:

```
sudo iptables --list
```

This will give you a list of rules. For each packet, the firewall starts at the top, and each rule is a pattern to match that might contain a port, an IP address, or some other way of matching a packet. Each rule has either an accept, reject, or log action. If the packet matches that rule, the firewall performs the action; if it doesn't, it moves onto the next rule until it gets to the bottom. If nothing matches, the firewall accepts the packet. Nftables works in much the same way.

Q OK. If iptables is so popular and nftables works in much the same way, why should I bother with nftables?

A That's an excellent question. Iptables is a mature, well-understood technology that you shouldn't be in a rush to get rid of, but at the same time, it does have some weaknesses. For example, iptables only works with IPv4, which is still the most common version of the IP protocol, but IPv6 is slowly becoming more used. There is an IPv6 equivalent (rather unimaginatively called ip6tables), but this means that you have to duplicate all your work to make sure you stay secure on a machine that supports both versions (and you should support both versions if possible).

The second major difference is that the rule syntax allows more concise expression. In many cases, rules that would have taken many lines of iptables rules can be done in a single line with nftables. This might not seem like a huge deal if you're only using small rulesets, but it's not uncommon for server setups to have iptables configurations that span several hundred lines, and shortening this makes it much easier to maintain.

Beyond this, there are a few advantages for larger setups, such as faster updating of rules, but these are only really applicable to large data centers.

In short, there are no major problems with iptables, but nftables is a bit better and likely to be the future of firewalls on Linux.

Q When can I get started with nftables?

A Right now [1]. Well, depending on your distro, that is. Support for nftables first came into Linux with kernel version 3.13, which was back in January 2014, so most modern distros should have it available (although you might have to install a package from the repository to get it to work) [2]. If you're using a slow-moving enterprise distro, you might have to upgrade to get support. Take a look at your distro's documentation for more details.

Q So, I guess this begs the question: Should I be using iptables or nftables?

A There's no simple answer to that. If you're currently using iptables and it's working for you, then there's no pressing need to switch at the moment. On the other hand, if you're not using a firewall, or learning how to configure them for the first time, then you might want to consider nftables, because it's likely to be the future of Linux firewalls. ■■■

Info

[1] Want to get up and running with nftables? The project website (*www.netfilter.org*) has more information than you could possibly want.

[2] Nftables has been part of the Linux Kernel since version 3.13, but it needs to be selected before compiling, so make sure to select the right options if you build your own.

Valentine Sinitsyn develops high-loaded services and teaches students completely unrelated subjects. He also has a KDE developer account that he's never really used.

CORE TECHNOLOGY

Look for intruders and study the health of your system with Linux auditing tools. BY VALENTINE SINITSYN

Audit Your Linux Box

No one enjoys being tracked. In Free Software and Linux, we take privacy very seriously. Yet, we sometimes set surveillance cameras to watch the back yard. We hardly ever look at the recordings, unless things go wrong. Then we could use videos to learn who broke that window.

Audit in Linux works much the same way. It captures security-related events, such as file access, system calls, user logins, or system reboots. Then it stores these logs safely and lets you search through them. This process doesn't add any security by itself, but it helps to track intruders. Having this is a prerequisite to Common Criteria certification, and it's a good way to peek into the system's operation for learning, fun, and profit.

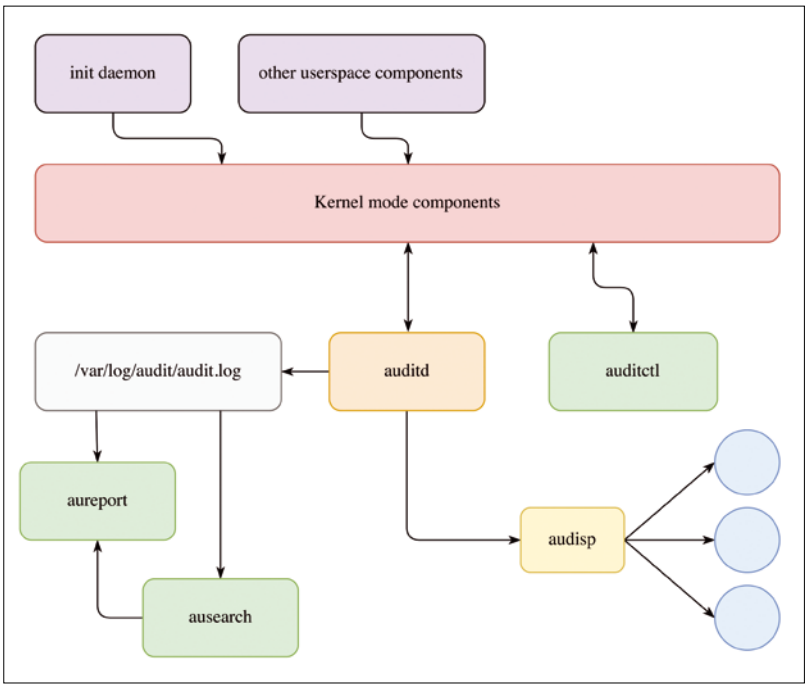


Figure 1: The Linux audit framework: major components and their interactions. Arrows indicate possible data flow.

The Big Picture

The Linux audit framework spans multiple components, both in userspace and in the kernel (Figure 1).

A story begins when something generates an audit event. For instance, when you open a file, the kernel can learn about it using the very same mechanism covered previously [1]. Or, if you want to trace system calls, the kernel raises a flag on a process descriptor, which causes Linux to fire an event when it returns from the call.

Not all audit events originate from the kernel, though. When the system boots, switches run levels, starts or stops daemons, etc., an `init` process such as systemd logs these life cycle events. This also happens when a user logs in or out. Technically, such messages end up in the kernel as well, so all other audit framework components have a single events source.

The kernel reports audit events via a Netlink socket. In userspace, a daemon called `auditd` reads them and stores them in audit log file. `auditd` can also forward events to a dispatcher daemon, `audisp`. This component acts as an event multiplexer for programs, which want to analyze events in real time, such as Intrusion Detection/Prevention Systems (IDS/IPS). You can also use `audisp` to send events to storages other than local disk, such as remote servers. This helps keep them secure. Even though audit logs are inaccessible to users other than root, an intruder with sufficient permissions can still tamper or destroy them.

Listing 1 audit.log Sample

```
type=DAEMON_START msg=audit(1478917138.665:1665): auditd start, ver=2.4.5 format=raw kernel=4.4.0-47-generic auid=4294967295
                                                  pid=406 subj=unconfined  res=success
...
type=SYSTEM_BOOT msg=audit(1478917138.740:5): pid=433 uid=0 auid=4294967295
ses=4294967295 msg=' comm="systemd-update-utmp" exe="/lib/systemd/systemd-update-utmp" hostname=? addr=? terminal=?
   res=success'
```

Many things are happening in your system at any given moment. Logging all of them would probably be overkill and a waste of resources. How does the kernel know which one to remember and which one to forget? Audit rules tell it. These rules define which files, directories, and system calls the kernel should monitor. You also use rules to select processes to trace or users whose actions you want to follow. `auditctl` communicates rules to the kernel, either during the system startup or dynamically in run time.

Now, with events recorded somewhere, how do you get a sense of them? `aureport` tool comes to the rescue. It produces a summary or detailed report upon your request. There are also ways to visualize these reports, as you'll see shortly. If you look for a specific event, use the `ausearch` tool. A common pattern is to find an event of interest in `aureport` (Figure 2), and then take its event ID and retrieve the details with `ausearch`.

Getting Your Feet Wet

For starters, let's look at how all these pieces work together using vanilla Kubuntu 16.04 LTS as an example. Once you boot the system, the audit log, which resides in `/var/log/audit.log`, should already have some events (Listing 1).

Listing 1 shows a so-called raw log format: `auditd` decodes kernel messages and persists them in the same form they arrive. There are no other options available out of the box, but you can ask `audisp` to redirect messages to the `syslog` if you wish.

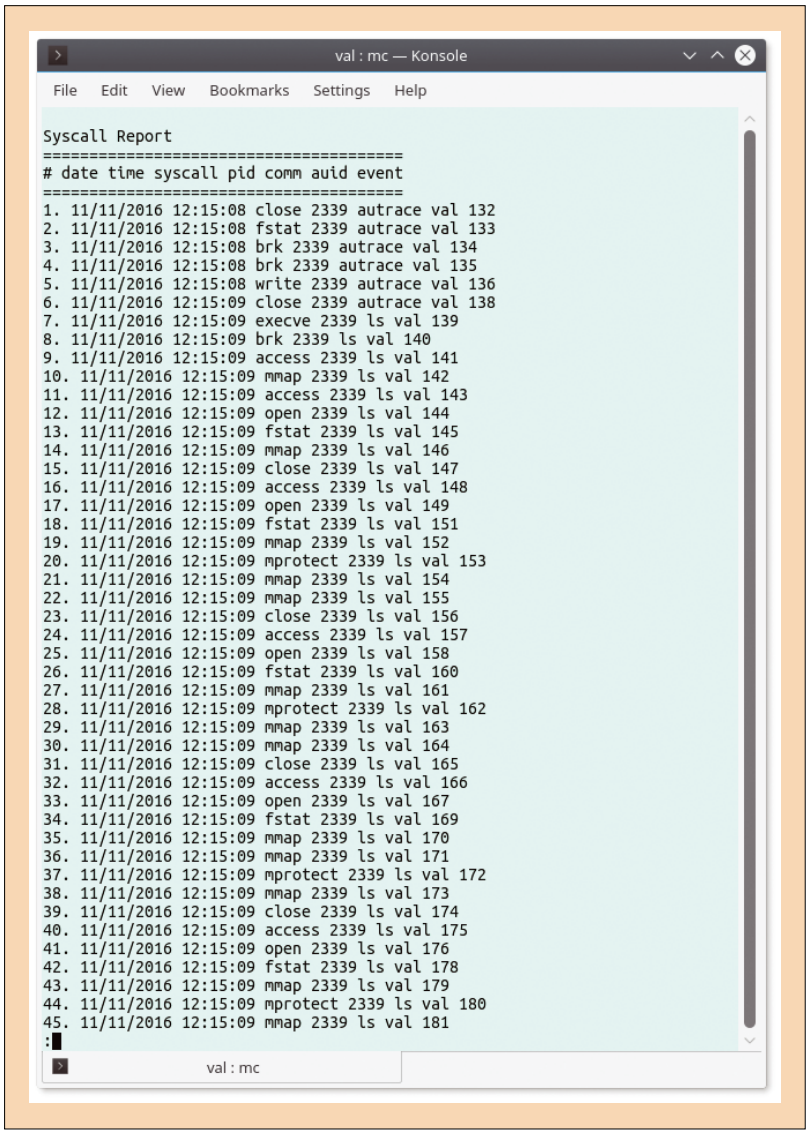
You may now wonder what happens if `auditd` is not running. It depends on the setting called "failure mode." If you are serious about security, the kernel may shut down the whole system. The default behavior is just to put a warning in the `dmesg` log, though. Similar options are available for when the system is low on disk space or a disk error occurs [2].

Despite the name, raw log format is mostly self-explanatory. As you see in Listing 1, each audit record consists of several key-value pairs or fields. `type` sets the record type. Suppose a `SYSTEM_BOOT` event occurs when the system boots, and `DAEMON_START` fires when `auditd` itself comes to life. Note it happens before the system is booted.

Different message types have slightly different fields yet many of them are common. Say, `msg` stores both the timestamp and the message ID, colon-delimited. To query a record by its ID, you can do the following (`#` denotes root command prompt):

```
# ausearch -a 1665 -i
----
type=DAEMON_START msg=audit
  (11/11/2016 18:18:58.665:1665) : auditd start,
  ver=2.4.5 format=raw kernel=4.4.0-47-generic
  auid=unset pid=406 subj=unconfined res=success
```

Figure 2: Sample `aureport` output, which shows a detailed system call report on a vanilla Kubuntu 16.04 LTS box.



Here the `-a` option tells `ausearch` the ID of the record you are interested in, and `-i` prescribes to convert numeric values, such as user identifier (UIDs) or timestamps, to their human-readable representations. Sometimes, this search yields more than one event record. Events related to a single operation (e.g., a system call) share the same message ID. Note that message IDs don't persist across system reboots.

You've probably guessed what `pid`, `uid`, `gid`, etc. are for. `auid` stands for Audit user ID and is also known as "login ID." You get one when you login, and it stays unchanged during the session. `comm` is the command, in kernel's parlance. As you see, it could be different from the executable (`exe`). `success`, of course, tells the operation status, and the `key` field acts as an administrator-defined tag; I'll show you an example.

To get an events summary, use the `aureport` tool (Figure 2). The command also accepts a number of switches to build reports for specific event groups. For example, `aureport --auth` reports authentication attempts (either successful or not), `aureport --file` tells about files access, and `aureport --comm` does the same for commands. Note that each detailed report is a table, normally having "event" as the last column. That's the ID you can use to lookup event details with `ausearch`.

`aureport` has some filtering capabilities built-in. You can limit the timespan with `--start` and `--end` switches. Both accept the date and the time (in this order) and are locale aware. Here, *11/12/2016* and *12.11.2016* are both valid, depending on which part if the globe you are, and refer to the same point in time. Shortcuts like *now*, *today*, *recent* (not more than 10 minutes ago), or *this-week* are also supported. You may also filter successful operations only with `--success` [3].

The `ausearch` querying capabilities are, as per name, more advanced. You can filter by message type or types (comma-separated), `pid`, command name, and user ID, including `auid` (aka login ID). Free-text search is available for string-based fields, such as file names, and `ausearch` is clever enough to match whole words if needed. The later means *systemd* matches *systemd*, but not *systemd-update-utmp*, for instance. You can also filter by date range or operation status the same way you do in `aureport`.

It is possible to pipe `ausearch` output into `aureport`, to make the latter read it instead of audit log. For example, this is how you build a neat summary for the events of interest:

```
# ausearch ... --raw | aureport --summary --file
```

Note the `--raw` argument to `ausearch`, which makes it output event records in raw form, as in the logs. (See the "Audit Beyond Commands" box for more information.)

Your Game, Your Rules

As you can see, the default configuration already captures some interesting things. Yet there are many possibilities for fine tuning. The Linux audit framework lets you install rules that define events of interest. Static rules reside in `/etc/audit/audit.rules` or `/etc/audit/audit.rules.d`. When `auditd` starts, its init script calls `auditctl` to load these rules, so they are always in effect. You may also use the `auditctl` command to install custom dynamic rules that you don't want to survive a system reboot. The syntax stays the same in both cases: You either store `auditctl` command-line switches in the rules file or supply them to the command directly.

`auditctl` is a multitool. It can do global configuration: disable audit, lock the subsystem so no one (even root) can change it, flush rules, or set the failure mode. It can also list rules or query audit status. And, of course, it can also manage your own custom audit rules!

Rules reside in one of four kernel lists and have one of two actions: `never`, which suppresses event generation, or `always`, which induces it. The most used list is `exit`, which stores rules to run at the end of a system call. The `user` list applies rules to events coming from the userspace. Recall that many userspace components, such as an init daemon, generate audit events, often unconditionally. This way, you can iron out these events. The `exclude` list is used to filter events by type. The last list, `task`, is rarely seen in practice.

Each rule also has a set of preconditions. They come in form of C-like comparisons (`=`, `!=`, `&=`, and so on) either against fixed values (`-F`) or between event fields themselves (`-C`). All conditions must be met for an action to trigger. Finally, you can attach an arbitrary string (called a "key") to a rule.

Audit Beyond Commands

Userspace audit tools rely on two libraries, *libaudit* and *libauparse*, to do the low-level stuff. These libraries constitute the public API. This means you can use them in your programs as well. This is not something unusual. For example, when an init daemon wants to generate a `SYSTEM_BOOT` event, it calls `audit_log_user_message()` or a similar function from the `libaudit`.

Administrators rarely code in C. They use scripting languages such as Bash or Python. For the former, userspace audit tools should cover most needs. If you code in Python, you'll be happy know that both libraries provide official bindings for Python 2 and 3. They are generated with SWIG and follow the native C API closely. There is no separate documentation, unfortunately: You have to consult `libaudit` and `libauparse` man pages directly. The good news is that these bindings should be available in your package manager as *python-audit* or something similar.

This helps to filter events produced by the rule: `ausearch -k` does just that.

Remember that only processes with `CAP_AUDIT_WRITE` capability (this means root in general) can send audit events from the userspace [4]. That's how you silence them for a while:

```
# auditctl -a user,never
```

The `-a` option tells `auditctl` to append a rule to the list; `-A` prepends it, and `-d` deletes. The order is important because rules are evaluated until the first match. Check that the rule was installed with `auditctl -l`. Now, if you generate a user-space event with `auditctl -m`, the kernel will happily ignore it.

Okay, this was an artificial example. Normally, you use `auditctl` to install filesystem rules, called watches, or system call tracers.

Watching the Filesystem

Consider the following scenario. You have a directory, say, `/var/cache/something` that keeps a cache expensive to rebuild. Yet something cleans up files in it on a regular basis. What is it? A cron script? An over-diligent user? Or maybe a malware? Stop guessing – the audit subsystem can tell you for sure (Figure 3).

You start with adding a watch for the pathname you want to monitor:

```
# auditctl -w /var/cache/something ➥
          -p w -k "cache offender"
```

The `-w` option tells the filesystem location to watch. The `-p` option describes access permissions of interest to us: reads (`r`), writes (`w`), or attributes changes (`a`). They are not standard permissions bits you operate with `chmod`, but rather syscalls issued at the filesystem object, or flags it is `open()`ed with. File removal is considered to be a directory write, so you track writes here. You also add a key to find generated events later.

Now, wait for the directory to become empty again. When this happens, delete the rule to save system resources, then run:

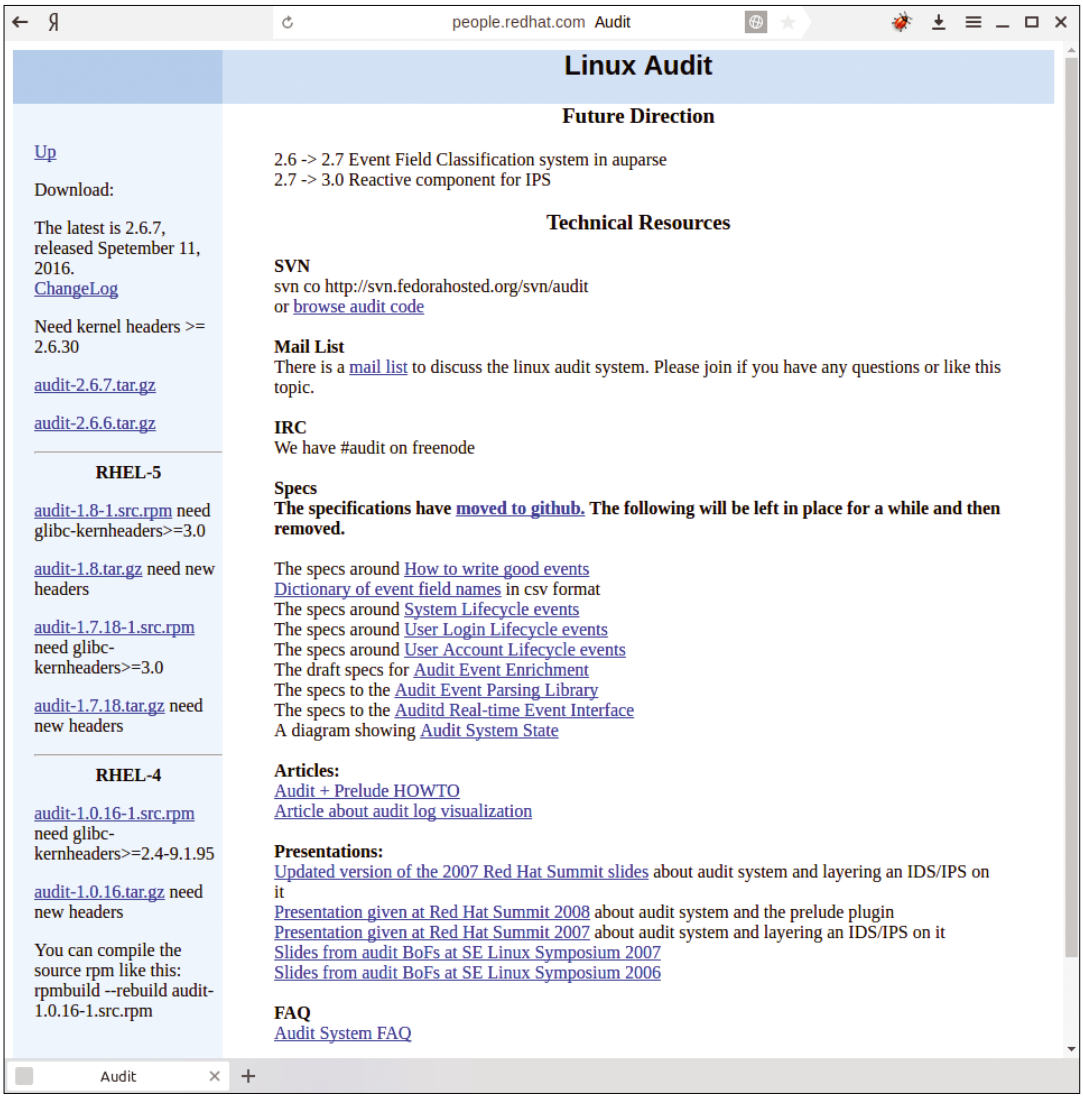


Figure 3: The Linux Audit homepage is an example of old-school web design, but it also contains many useful specs.

Listing 2: Syscall event record. a0 is the pathname

```
# ausearch -m SYSCALL --syscall unlink -i
...
03 type=SYSCALL msg=audit(11/14/2016 02:30:37.670:236) : arch=x86_64 syscall=unlink success=yes exit=0 a0=0x560e36a15250
                                                         a1=0x0 a2=0x0 a3=0x8 items=2 ppid=1 pid=2198 auid=unset uid=root
                                                         gid=root euid=root suid=root fsuid=root egid=root sgid=root
                                                         fsgid=root tty=(none) ses=unset comm=cupsd exe=/usr/sbin/cupsd
```

```
# ausearch -k "cache offender" -i
```

As an alternative, you can filter by the directory's pathname:

```
# ausearch -f /var/cache/something -i
```

The output could be lengthy, but it will show you a command, a user and process ID, and everything else you need to reach the offender and explain that he is doing something wrong.

Tracing System Calls

Watching the directory as above is the most straightforward solution, but not the only one. Another way is to trace `unlink(2)` and `unlinkat(2)` system calls. These syscalls are used to delete a file, and once you have relevant event records, `comm` and `pid` will tell you who is doing this.

Adding a syscall rule is not much different from installing a filesystem watch. Remember however that all userspace processes on your box issue system calls and do this quite often. So, if you enable system call tracing, you get a performance hit. In our case, the kernel will have to iterate over rules in the exit list even for processes which never issue `unlink(2)`. The fewer rules you have, the smaller the penalty is. Match multiple system calls in one rule whenever possible, as in the snippet below:

```
# auditctl -a exit,always -F arch=b64 ➥
  -S unlink,unlinkat -F path=/var/cache/something ➥
  -k "cache offender, part two"
```

Two things to note here. First, I had to specify the system's architecture. Linux distinguishes system calls by their numbers, and these numbers may overlap on different architectures, including 32- and 64-bit x86. Here, I simply force a 64-bit system call table. Second, I use a pathname filter to ignore events unrelated to the cache directory. `unlink(2)` and `unlinkat(2)` also receive the pathname as an argument, but you can't filter by it. The reason is you only get pointers, not string values in the audit log (Listing 2).

When the cache empties again, make a similar search in the event log:

```
# ausearch -k "cache offender, part two" -i
```

Of course, it should yield the same offender as before.

Sometimes, you want to trace syscalls in one specific command. Sure, you can add a rule, run the command, and delete it afterwards. The `autrace` tool wraps this sequence for you. Conceptually, it's similar to `strace`, yet it produces audit events instead of console output.

Note that `autrace` flushes audit rules if you have them installed. That's because the tool adds its own temporary rules, and it doesn't want to mess up your rules. Otherwise, using `autrace` is straightforward (Listing 3).

The output shows the command you can use to dump the trace. `autrace` also runs in a resource usage mode [5]. ■■■

Listing 3: Running autrace

```
# autrace /bin/true
Waiting to execute: /bin/true
Cleaning up...
Trace complete. You can locate the records with 'ausearch -i -p 2685'
```

Info

[1] "Filesystem Monitoring" by Valentine Sinitsyn, *Linux Magazine*, Issue 194, pg. 74.

[2] auditd.conf(5) man page: *https://linux.die.net/man/5/auditd.conf*

[3] aureport(8) man page: *https://linux.die.net/man/8/aureport*

[4] capabilities(7) man page: *https://linux.die.net/man/7/capabilities*

[5] autrace(8) man page: *https://linux.die.net/man/8/autrace*

[6] mkgraph home page: *http://people.redhat.com/sgrubb/audit/visualize/mkgraph*

[7] mkbar home page: *http://people.redhat.com/sgrubb/audit/visualize/mkbar*

Command of the Month: mkgraph & mkbar

The Linux audit framework collects a lot of data – how do you get a sense of it? A picture is worth thousands of words and visualizing audit data would certainly be helpful.

Two simple scripts are available for these purposes: mkgraph [6] and mkbar [7]. Download the scripts and make them executable. The scripts are quite old and may need some tweaks to run on your box. If the commands below yield empty files, remove the `-t` switch to `read` commands inside the scripts. You may also want to adjust `$EXT`, which defines the output file format.

You'll need Graphviz for `mkgraph` and Gnuplot for `mkbar`, so install those from your package manager now. Comments in the scripts give you some usage examples. This is how you see which programs run which system calls:

```
LC_ALL=C aureport -s -i | ➲
  awk '/^[0-9]/ { printf "%s %s\n",$6, $4 }' | ➲
  sort -u | ./mkgraph
```

This uses `awk` to get the fourth and the sixth columns (syscall and comm) from the detailed system call report. Then, it removes duplicates and feeds the output to `mkgraph`. You can see the result in Figure 4. `mkbar` works the same way, but it draws bar charts, which are quite useful for quantitative metrics such as event count.

There are ways to build visualizations that look better. The `mkgraph` and `mkbar` tools aren't meant to be ready for your investor presentations, but they are nifty little tools you could use in your day-to-day work. ■■■

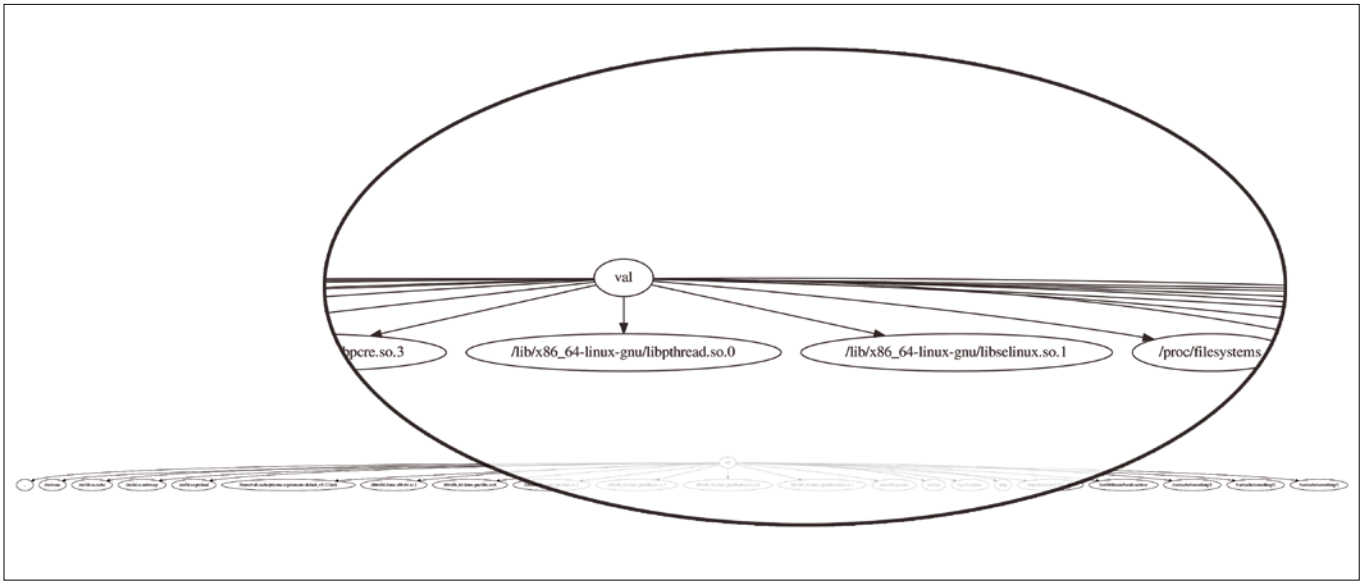


Figure 4: An `mkgraph`-generated flowchart showing some files a user has accessed. Not really aesthetic, but sometimes useful.

FOSSPicks

Sparkling gems and new releases from the world of Free and Open Source Software

Graham tears himself away from updating Arch Linux to search for the best new free software. BY GRAHAM MORRISON

Image processor

Darktable 2.2.0

Darktable is a photo management and processing tool that's good enough for professional use and can even replace costly proprietary alternatives like Adobe Lightroom. It's brilliant because it allows you to split the tasks of importing, processing, and rendering photos into logical steps, and it provides a lot of cutting edge tools to help you. The color processing, exposure adjustments, lens correction, and sharpening tools are essential, for example, and have often compensated for my very poor photography skill, even when those photographs are destined for print.

This is a major release with lots of major new features – there have been more than 2,000 commits to the project's repository since version 2.0, for instance. But, if you play with only one new feature, is has to be the perspective tool. Ever since the days of Deluxe Paint, there's been nothing more satisfying that transforming a two-dimensional map of pixels through three-dimensional space. Rather than remapping pixels across a simple polygon, however, Darktable allows you to compensate for perspective distortion effects, letting you adjust the vertical and horizontal lens shift, for example, as well as shear and rotation within this 3D transformation. But – and this is the clever part – it does this by first analyzing your photo for structural features within the image itself, looking for parallel linear elements to use as anchors for the transformations. These create converging lines to a vantage point, which is then used to calculate the perspective and how the image should be transformed. The results are natural and brilliant. You can make walls vertical, for example, and you can shear the image around one of the diagonal lines within the image. Perfect if you don't have a quadcopter handy.

1 History and Snapshots. Changes to your photos in Darktable are never permanent. **2 Preview.** Thanks to OpenCL, many processes are hardware accelerated and update in real time. **3 Workflow.** More than an editor or a library, Darktable manages your image workflow. **4 Processes.** Modular effects are enabled or disabled and applied serially. **5 New Effects.** Drag new effects into your process stream. **6 Management.** Tag and organize vast photo collections. **7 Perspective editing.** The best new feature in this release. **8 Metadata.** You can even edit the location where a photograph was taken.

Another new tool is *liquify*. This groups together a series of other tools into one easily controllable effect. The effect is controlled by drawing a point, a line or a curve onto your image, and the way each affects the pixels is slightly different but generally looks like the image is printed onto a moving piece of material. It's great for working with text, for example, by warping its shape. Our next favorite new module changes the colors within an image by swapping them with colors held in a table. It's called the "color checker lut module," with "lut" meaning look up table. A simple lookup table is how some of the color rendition presets work, swapping out one palette for a new one and replacing one set of colors with another, but with the new plugin you can now create your own replacement palette or adjust others to nail the effect you're looking for, rather than relying on presets.

Darktable is complex and slightly overwhelming for any beginner, but it's now offering features rivaled only by its expensive proprietary competitors. With many smartphones bundling half-decent cameras, and some even capable of full frame RAW output, there's never been a better and more affordable time to get into photography. Tools like Darktable give you the power that only professionals had access to five years ago, making this update and the continued success of the project even more impressive. If you've only ever tried Shotwell, you owe it to yourself, and your photographs, to give Darktable a try. ■■■

Project Website
https://www.darktable.org/

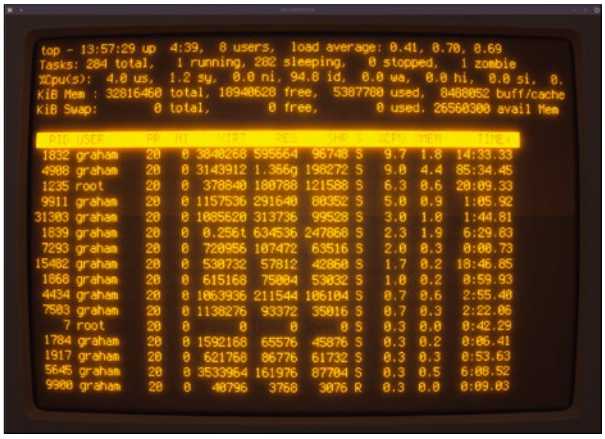
Terminal Emulator

Cool-Retro-Term 1.0.0

It wasn't so long ago that the quality of your monitor was judged by how flat the screen could be made, and how close together the cathode ray tube could fire photons to produce as many colors as possible. But now that we're in the "post CRT" era, we're inundated with flatness and tight pixels. This may be why curved monitors are the latest thing, and also why Cool-Retro-Term exists.

Cool-Retro-Term is a wonderful terminal emulator, in the truest sense. While hosting your terminal session, it emulates many of the physical characteristics of ancient displays, rendering one-time flaws in real time on your modern GPU. The default profile, for example, recreates one of those lovely amber VT220 terminals, complete with screen curve, flickering, persistence, and glowing interference. Different profiles switch from this to the green of an Apple][display, or the misfiring rays of some random IBM PC-AT clone. The authenticity of the appearance is uncanny, and takes you back 20 and 30 years if you're old enough to have experienced the real hardware.

All of these effects are implemented using a selection of shaders with some excellently chosen fonts. The shader values can be changed, and they include bloom, curve, static burning, glow, jitter, ambient light, flickering, and RGB shift. When combined with different fonts and colors, you can recreate the ancient displays of almost any hardware and save this as a new profile. Unless you're a film director working on a sequel to *War Games*, it's not going to make you more productive, but it will definitely make you smile. ■■■



Based on Qt QML and qtermwidget (Konsole), Cool-Retro-Term is as good in operation as it is in looks.

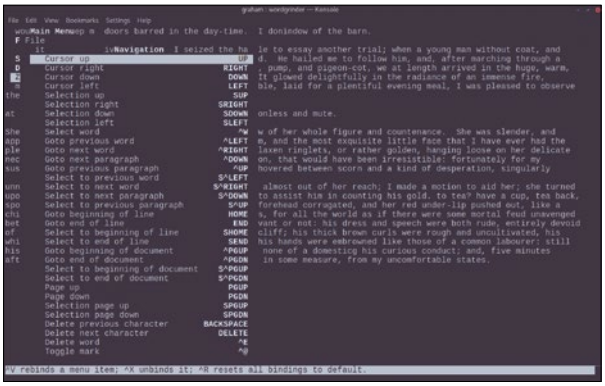
Project Website
https://github.com/Swordfish90/cool-retro-term

Terminal word processor

WordGrinder 0.6-1

Back in the age of early CRT monitors, letters, reports, and doctoral theses would all be produced using a word processor with a text interface. There was no graphical interface. The world's most terrible acronym, WYSIWYG, was still a secret, and regardless of whether a word was italic, bold, or written in Comic Sans, the text would look just like any other text on the screen. Instead of changing these styles visually, old-fashioned word processors would use inserted codes and shortcuts. This made them harder to use, but crucially, it also made them lightning fast and efficient, just like using vim over *gedit* or *kate*. So, it's not surprising that there are still WordStar aficionados, for example, clinging to their DOS word processor almost two decades after its last release.

That's why it's so good to see a new(er) addition to this venerable genre. WordGrinder is a word processor for the command line, rather than a text editor for the command line. It's been developed to create text documents that you'd normally create in LibreOffice, only from within a fast and efficient text interface. It will even output the same formats, including ODT, alongside more text-centric formats like HTML, Markdown, and LaTex. The always on-screen word count and the flat menus, file requester, and search and replace panes, make this more friendly than vim and better for letters than nano. It's also an excellent option if you happen to have a Raspberry Pi working as a USB print hub and want to quickly create a good-looking document you can print and send without having to run a graphical interface, or even connect to a machine locally. And if you're still using WordStar in DOSBOX, perhaps you can finally reconfigure that muscle memory. ■■■



It may be a simple command-line word processor, but it's comprehensive enough to include Unicode support.

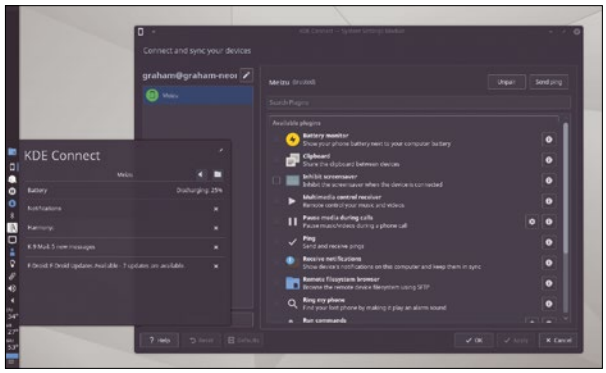
Project Website
http://cowlark.com/wordgrinder/index.html

Mobile notifications

KDE Connect

I've mentioned KDE Connect before, and recent developments mean it's becoming one of KDE's best features. KDE Connect is an applet and background service for your desktop, but it's also an open source app for your Android phone. Installed either via Google Play or F-Droid, the Android app connects to your desktop whenever both your computer and phone are connected to the same network, although you do need to accept a PIN first for authentication. After both devices are connected, which happens automatically and transparently after the first synchronization, you can now see lots of information from your phone and perform lots of new actions. Things you can see include battery status, any notifications from your phone (enabled separately), and the phone's filesystem. You can transfer files, send and receive files in either direction, and make your phone ring if you need to find it. You can use your phone as both a keyboard and a touchpad, which is an excellent option for presentations or remote control, and you can share the "copy and paste" clipboard with either device or PC, which is great for URLs and passwords. You can even run commands remotely, which is good for shutting down your PC from the kitchen.

New in the latest release, you can also reply to SMS notifications from your desktop. This feels like the modern age of integrated devices, and it's genuinely useful. It means you can keep your phone plugged in somewhere and still send messages and access its core features. To get this feature to work, I needed to delete all our old configuration files in the `~/.config/kdeconnect` directory and re-sync devices, but this could have been because I'd iterated over many old versions during development. After a restart, KDE Connect is faultless and highly recommended. ■■■



KDE is now synergizing with your phone by letting you send SMS messages from your desktop.

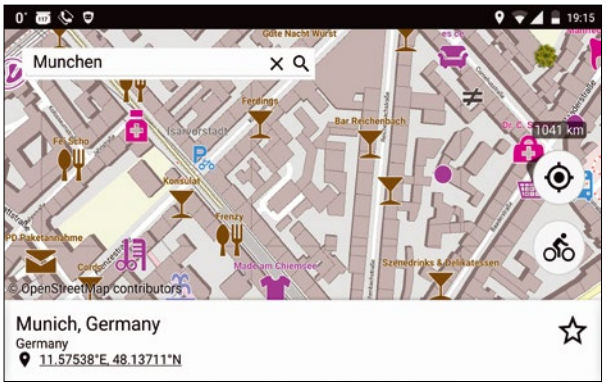
Project Website
https://community.kde.org/KDEConnect

Android navigation

Marble Maps 1.0

It could be argued that Android is the most widely used and distributed version of Linux. And even though Google is seemingly fighting to make its new Android frameworks as proprietary as possible, its Android core remains open source and at the heart of other mobile open source operating systems. This may be why some Linux projects are expanding into Android development as an extension of their more traditional roots. The KDE project is one of those pioneering spirits. KDE Connect (above) links your phone to your desktop, while Marble Maps takes this further, testing KDE's long-promised platform agnosticism by porting the Marble libraries to Android and creating a whole new app for browsing OpenStreetMap data in the process.

Marble Maps itself is a navigation app that uses the same data and rendering routines as its desktop counterpart. This is evident from the first launch, where the vectors are rendered as finely as your phone's display will allow. On any screen with a high DPI, it looks amazing. It's also quicker than the standard open source OpenStreetMap app, OsmAnd, but it's also bereft of many of its features. There's the same routing planning you get at http://www.openstreetmap.org, but very little else. No control over caching, no plugins, no tracking, no importing – none of the features that make OsmAnd so useful. But this is just a first release, and because of the beautiful rendering and quick screen update, we'd still recommend Marble Maps as the app to use when you're lost in a city, just as long as that city is in Germany, where there's even 3D building data, or the rest of Europe or the USA where OpenStreetMap details are highest. ■■■



Marble isn't the only KDE app making its way to Android; there's also a version of KStars, too.

Project Website
https://marble.kde.org/

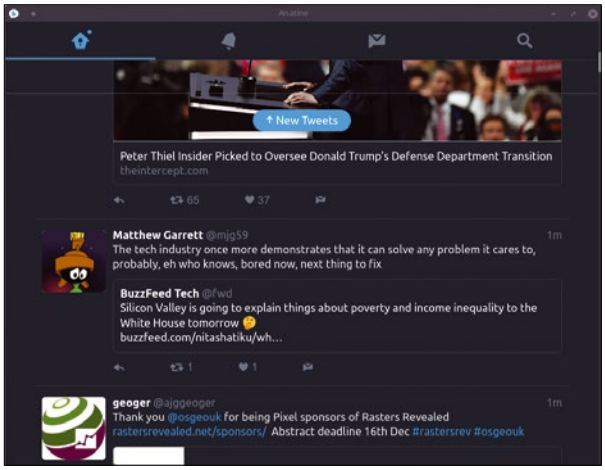
Desktop Twitterer

Anatine 1.1.0

If there's anyone still brave enough to look at Twitter, it remains a social media platform that has its uses. It's unrivaled for its ability to capture news events as they unfold, for example, and there are still many Linux developers and open source advocates sharing their thoughts with the world via its 140 characters. But the initial glut of Linux clients that followed its launch way back in March 2006 have mostly dried out. It's seems most users are happy to use the now permanently web-bound Tweetdeck or stick with their old Twitter clients.

But there's also Anatine, which is definitely worth a look if you need a desktop client. Anatine isn't strictly a desktop application, though. It's built around an encapsulation of the mobile twitter site (mobile.twitter.com) using the Electron Node/Javascript and Chromium framework, which means it's basically a browser loading a single page. This also means you can customize the experience as if you were using a single application, and what's particularly good about Anatine is its use of keyboard shortcuts and a rather excellent dark mode implemented via easily editable CSS.

Anatine can be run directly from the download (although it's a zip bomb, so be careful) and needs a single login to get going. This is because it's using an official web portal, rather than accessing your account via a limited access API, which is what destroys many third-party Twitter clients. When logged in, Anatine is simple, quick, and looks great on-screen. It doesn't have the power of Tweetdeck, but neither does it eat up the same amount of RAM and resources, and it lets you easily keep on top of your Twitter updates or ignore them completely. ■■■



Filter out the hate, and Twitter is still a good way of keeping in touch with the open source community.

Project Website
https://github.com/sindresorhus/anatin

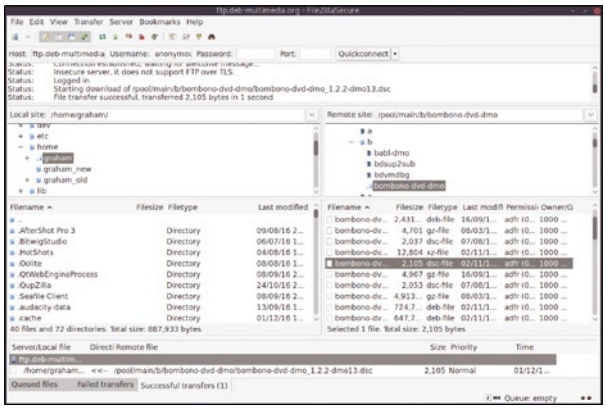
FTP client

FileZilla Secure

There are two reasons why the FileZilla fork – FileZilla Secure – is brilliant. And, the first is that it's not hosted on SourceForge. SourceForge was the GitHub of its day. It hosted the websites, the binaries, and the code for many of the projects we took for granted, including my own. But changes of ownership at SourceForge and commercial pressure changed what initially seemed like an altruistic initiative into one that was riddled with advertising. Even worse, the advertising even made its way into some of the binaries as malware, seriously affecting the site as a source for untainted open source. One of the most important projects hosted on SourceForge was FileZilla, a perfunctory but powerful graphical FTP application.

When someone asked for an application to make FTP transfers easy, especially on Windows or OS X where free alternatives don't exist, FileZilla was the recommended application. But because FileZilla was hosted on SourceForge, it became a difficult recommendation.

The second reason why FileZilla Secure is brilliant is that, despite many requests, FileZilla (the original one) still stores your password as plain text. Plain. Text. That means that anyone with any kind of access to your computer, including malware that's been secretly installed, can take your passwords and easily use or sell them. FTP security is bad enough, but this makes it considerably worse. The solution is trivial for a programmer to implement, and that's exactly what's been done with FileZilla Secure. It's a fork of FileZilla that isn't hosted on SourceForge and that encrypts your passwords. As a bonus, FileZilla Secure has also upped the number of threads that can be used when transferring data, potentially increasing transfer rates by five times. ■■■



Don't risk using the original FileZilla; now there's a secure fork.

Project Website
http://www.filezillasecure.com/

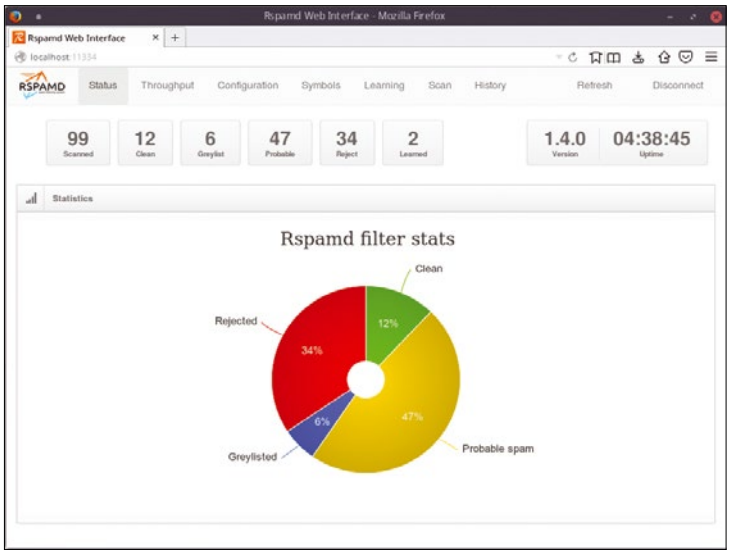
Spam Filter

RSpamd 1.4

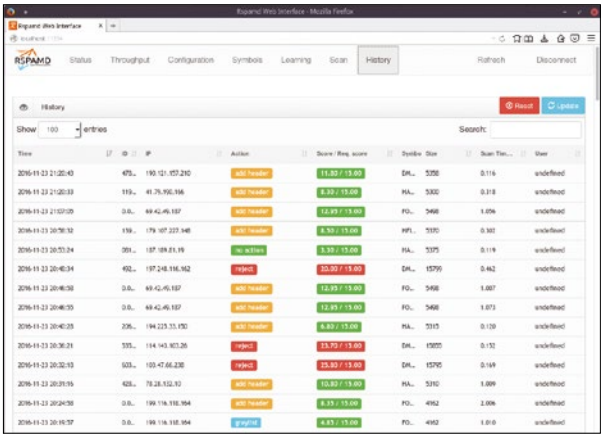
For the past three years, I've been the maintainer of the Linux Voice mail server. Over that time, as you might expect, it's taken a hammering. And as anyone who runs a mail server will know, that hammering is thanks to the 9999:1 spam to real email ratio. Fighting spam is a full-time job in itself, and one that equally consumes your time, your computing resources, and network bandwidth. It's partly what makes hosted email services like Gmail so attractive as Google uses its vast network of data to filter out spam automatically. But, even with Google's resources, spam emails still get through.

For many years, the vanguard of open source spam fighting has been SpamAssassin, an email filter that runs on the mail server in conjunction with a Mail Transfer Agent (MTA) like Postfix. SpamAssassin uses a variety of clever techniques such as online blacklists, Bayesian filtering, and machine learning to mark and drop email before it hits your inbox, and considering the scale of the fight on its hands, it's very, very good. But SpamAssassin has started to feel a little neglected lately. Updates have slowed and more spam is getting through, and it's still difficult to configure and maintain. This isn't always SpamAssassin's fault though. It does the job so well that a spammer will often use a local installation to run their emails through, tuning their content until they can squeeze past the latest filters. As a result, I've kept an open mind about finding an alternative. And, after coming from nowhere, RSpamd appeared untested on the horizon. After the release of a new major update, version 1.4, it seemed the right time to finally jump the SpamAssassin ship and give something else a try.

Installing a new spam filter into a complex Postfix and Dovecot email system is never going to be easy, and installing RSpamd isn't easy. It uses its own Rmilter to talk to Postfix and stores data using Redis – none of which will be typically installed on the average mail server. But installation is easier than SpamAssassin, and the final part where you open your dreaded Postfix configuration file is actually the easiest part. After that, everything just seems to work. That's what's most impressive. Even with no email training, I found the default options in RSpamd to be slightly more effective from first installation than my SpamAssassin configuration with three years of heuristics.



RSpamd has a web user-interface, which offers valuable feedback that SpamAssassin can't touch without installing something like MailScanner.



For us, RSpamd discovers more spam than SpamAssassin, and performance is better too.

RSpamd also includes a very helpful web UI. This gives immediate feedback on how many emails are getting through, how many are blocked, greylisted, or marked. Greylisting is a delay and a resend request, letting the filter make a deferred decision on its status, whereas marked email are probably spam and are flagged as such in their headers. Thresholds for each level can be easily changed, and you can dive into the details for each filter through the web to create a configuration that works best for you. You can even use the web UI to help RSpamd learn from those emails that get through, as well as scan emails manually.

Thanks to a module system that uses LUA scripts, you can create your own filter modules or modify the included modules to suit your needs, and you can even import your own SpamAssassin filters, which will be huge benefit if you're thinking of migrating to RSpamd. Which is exactly what we've done. It may be early days, but so far, the results are excellent. ■■■

Project Website
https://rspamd.com/

NES Emulator

LaiNES

It's taken many years, but for once Nintendo has listened to its audience and created something most of us would never have thought possible – a recreation of the classic Nintendo Entertainment System. It's a smaller, self-contained console that can't take those old cartridges, but it's packed with 30 games and includes the controller. And it's even running Linux! For once, those of us with a hankering to play the many classic NES games of the 80s don't have to resort to emulation.

But that doesn't mean emulation is dead. It really means the opposite as Nintendo is obviously using emulation within its embedded Linux system, and developers are still creating new emulators such as LaiNES. What's remarkable about LaiNES is that it's a cycle accurate NES emulator written in approximately 1,000 lines of C++ code. Even though this total doesn't include a couple of libraries, it's still a hugely impressive accomplishment. So, too, is the way the emulation is cycle accurate. This means the cycle iteration of each chip within the old NES, and the way those chip cycles were clocked against each other, is perfectly emulated. It makes the emulator much more CPU intensive, but it also makes it much more accurately and theoretically capable of recreating all the same quirks and glitches that many games designers used as features back in the day.

It's size is evident in the speed the source takes to compile – under two seconds on our system,

LaiNES is resource hungry, taking around 40 percent of our i6700k CPU, but the main reason for this emulator is the source code.

and the resulting binary loads instantly. A simple GUI lets you navigate to your games, scale the view to either double or triple the original resolution, and change which keys are used for the two controllers. After that, the emulation just works. Select your game and play! ■■■

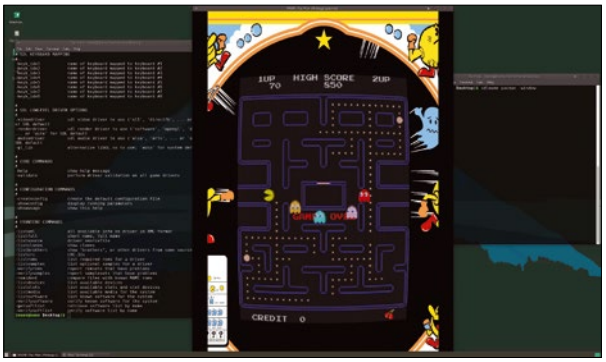
Project Website
https://github.com/AndreaOrru/LaiNES

Arcade Emulator

MAME v0.180

This is one of those projects with a crazy version number. 0.180 doesn't mean it's nearly a fifth of the way to a major release. That would make a mockery of its 20 years of development (MAME celebrates its birthday on February 5, 2017). Instead, it's a hard-fought testament to the many, many releases over those years, each adding too many platforms, too many games, and too many features to list. MAME has grown from an emulator for playing basic arcade games from the early age of computing, to a system capable of playing more than 10,000 different games. Since 2015, it has also included the computer and console emulators from MESS, and 2016 brought a much more functional internal UI for selecting games. But it's the last couple of releases that have got us excited, as these are the first to incorporate native OpenGL shaders for Linux.

The Windows version of MAME has always benefited from graphical effects coded to work with Microsoft's DirectX, and although OpenGL shaders have been created by users for Linux, they were never part of the main project. They instead required you to build them manually and add them to configuration files. The recent release of MAME has finally rolled these into the main project, and while you can't yet save every parameter, you can access dozens of effects for changing how MAME looks from the new UI. These effects include perspective distortion to change the angle of the screen or the curve of the CRT. You can add noise bars, distortion, and scanlines, change the beam



Party like it's 1984 with the latest OpenGL shaders built right into MAME.

scatter algorithm for the CRT and the convergence of red, green, and blue pixels, as well as play around with the color, gamma, and contrast. With a bit of practice, you can recreate the characteristics of almost any ancient and heavy screen, and MAME even includes presets. These can be pasted into a game's specific configuration file so you can have one set of shaders for Space Invaders, for example, and another for Rastan Sega.

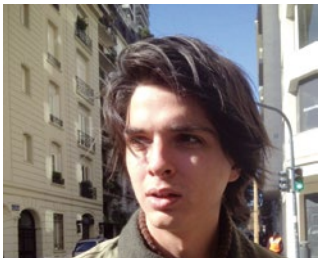
A little like archive.org, this is a project about documenting the history of computing and gaming. Of course, there will always be those of us who run a MAME-based arcade machine, playing games we couldn't afford to push coins into as children, but that's really a side show from the incredible breadth of MAME. ■■■

Project Website
http://www.mame.net/

GAMING ON LINUX

The tastiest brain candy to relax those tired neurons

BY MICHEL LOUBET-JAMBERT



Michel Loubet-Jambert is our Games Editor. He hasn't had a decent night's sleep since Steam came out on Linux.

There's plenty of Vulkan API news making the rounds, which is always good to see. The Radeon-Vulkan (radv) driver has been making plenty of progress, with lots of new features and bug fixes being pushed out, though still not really that stable as of yet. Along with that, Mesa 13 has seen upwards of 30 percent performance improvements on Vulkan over the last version on the Intel driver. At the same time, the porting house Feral Interactive has requested that Canonical provide a PPA for more updated Mesa drivers in order to be able to support more Intel and AMD hardware, which have unfortunately been neglected of late by many developers for such reasons. One interesting non-official project is VK9, which seeks to implement Direct X 9 over Vulkan, which may be useful to get more performance out of things like Wine. It would be nice though if this were done with d3d11, because it could hasten compatibility with a whole generation of games that are currently unplayable on Linux.

On the virtual reality side of things, the open source platform OSVR has been added to the supported platforms on Steam. This is a pretty big step for the project, because if it is to stand much of a chance commercially, then being on the biggest digital distribution platform is a must. It's also interesting to see that Valve isn't shutting out platforms that can potentially compete with Vive, and convergence with SteamVR has been hinted at in the past.

Deus Ex: Mankind Divided

Cyberpunk dystopias with political intrigue.

Web: *http://store.steampowered.com/app/337000/*
Price: £39.99/$59.99/EUR49.99

The series has benefited from the latest generation of graphics, bringing its world to life.

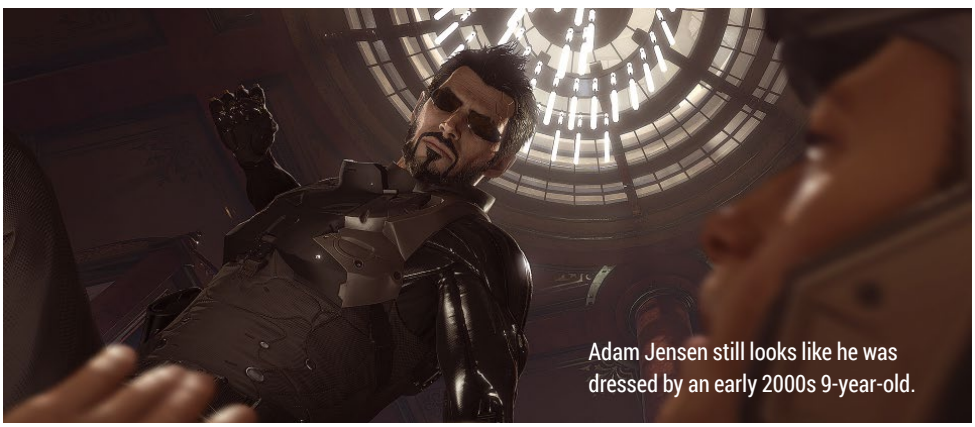
It's always great to see when big triple-A titles make their way onto Linux and even better when it's soon after the original Windows release, especially one so representative of the history of PC gaming. However, Mankind Divided really shows how far the platform has come since 2000, and even from 2011, when Human Revolution was released.

This game can be seen more as a direct sequel to Human Revolution, which is made abundantly clear in its very long sequence that summarizes the events in the two years prior to Mankind Divided. It is also made very clear that there's a lot going on in the game's dystopian future, from trans-humanism and mega corporations to apartheid and the Illuminati (not kidding). Don't let the name of the bad guys fool you, though, the story and fleshed-out world are incredibly enthralling and feel very well thought out, rather than something slapped together.

The graphics for this game are truly outstanding, and it's hard not to stop and admire the tiniest of details to appreciate the work that went into them, with these details being overwhelming at times. Choosing Prague as the main setting really lets the graphics shine, as having one of the world's most beautiful cities clad in cyberpunk decor looks both fantastic and eerie.

As for gameplay, it's hard to pin down, though perhaps best described as a stealth-shooter RPG. The action-focused parts of the game need not be action at all, as almost everything can be solved in various ways, be it through stealth or guns blazing, or discovering secret passages and using skills like hacking and cybernetic enhancements to progress. The story progression is also non-linear, with plenty of side quests that go above and beyond simple fetch quests and their discovery rewards exploration.

Mankind Divided does almost everything right. However, because its predecessor is not available on Linux, it is made a far more rewarding experience by doing some reading into its world beforehand.



Adam Jensen still looks like he was dressed by an early 2000s 9-year-old.

Transport Fever

A modern day answer to Transport Tycoon.

Web: *http://store.steampowered.com/app/446800/*
Price: £26.99/$34.99/EUR31.99

Both maps offer all vehicle types, but the European trains are faster and more advanced.

If playing OpenTTD feels a little dated, then Transport Fever certainly brings the transport management genre into the 21st century. Itself a sequel to Train Fever, it builds on those foundations to bring something far more refined and all-encompassing to the table.

The game offers two map flavors in Europe or the United States, along with their corresponding vehicle types, which consist of road, rail, air, and fluvial transport. The overall aim is to avoid bankruptcy while connecting together primary resource production with industries, building a supply chain to bring goods to consumers, as well as providing said consumers with a public transport network.

As the game progresses from the year 1850 onwards, technological advancements add both speed and capacity to the network, with towns and industries growing accordingly.

Although on paper this all seems fantastic, certain things can be wonky, such as the production chain randomly stopping and then not starting again, though these issues are being ironed out through patches. The campaign modes are also a little lackluster and cease to serve their purpose once the basics of the game are learned. Overall, the game is an extremely enjoyable experience and deserves to be called Transport Tycoon's spiritual successor.

Total War: Warhammer

A match made in heaven?

Web: *http://store.steampowered.com/app/364360/*
Price: £39.99/$59.99/EUR59.99

Being able to control large armies gives battles an epic sense of scale.

The combination of the Warhammer and Total War franchises makes so much sense that it's a wonder it's never been done before. With the Total War games having perfected Real Time Strategy combat and mechanics, Warhammer provides an established setting for the series to ditch historical accuracy and do something a little different without having to re-invent the wheel. Those large tabletop armies seem to mesh perfectly with the real time battles of the Total War franchise.

The game has a number of campaign modes, each revolving around different races of the Warhammer world. Each race provides a different backstory, as well as race-specific advantages and challenges. Far from superficial differences, these essentially force completely different styles of play, both on the battlefield and in the overworld. As a result, every campaign feels new and as entertaining as the last, despite taking place on the same map. Unfortunately, not all the races and factions are covered, although the base game does provide tons of content. Though the main thrust of the game is the battles, the overworld system is also fairly well fleshed out, with economy, building perks trade, and diplomacy to manage, making it a broad and detailed game overall.

ALSO RELEASED...

Football Manager 2017
The popular sports management franchise is back with a new iteration, featuring over 2,500 football clubs and over 500,000 players. As usual with these annual sports series though, don't expect anything groundbreaking since most changes from the previous version consist of roster updates and nothing essential for those already enjoying its predecessor. For those new to the game though, there's plenty of detail and immersion to be found.

http://store.steampowered.com/app/482730/

Silence
This sequel to The Whispered World does away with many of the point-and-click genre's staples to focus on story. The inventory system is gone, as is the fairly open world and dialog trees, leaving instead puzzles "on rails," which unlock the next part of the story. The art style is fantastic, seamlessly blending 3D characters with pre-rendered backdrops, whereas the simple gameplay is perhaps a sign that the genre is evolving.

http://store.steampowered.com/app/314790/

Motorsport Manager
If the career mode on racing games seemed like the most interesting part, then this is the game for you, especially considering many racing games have now cut that part out. It's certainly one of the best management sims out there, allowing the player to manage the financial and sporting performance of a racing team. Through its intuitive interface, it also manages to offer plenty of detail without being cumbersome.

http://store.steampowered.com/app/415200/

Intrusion Protection: Digital Self-Defense

No computer security is perfect, so make sure you've got a second line of protection.

BY BEN EVERARD

Let's take a look at two ways of making sure that, in the event of some bad guys (who may or may not work for a government) break into your machine, you find out they're there before they do any damage. The two methods differ in what they monitor: We'll use a host-based intrusion detection system (HIDS) to keep an eye on what's going on inside our machines, and we'll use a network intrusion detection system (NIDS) to try and detect suspicious traffic. Neither is foolproof, but these two complementary systems can significantly increase your chances of staying safe.

OSSEC – the HIDS we'll use – is built from the ground up to monitor multiple computers in a network. Years ago that might have only been relevant for people working on enterprise IT, but these days most people have many devices connected to their home network, and with the Internet of Things starting to take off, the number of computing devices in the average home is only likely to increase. Tripwire (see "Tripwire" box for details) is another option that's a bit easier to set up but only really suitable for one machine at a time unless you buy the enterprise additions.

Before installing OSSEC, you'll need to decide which machine should be the master. If you have a home server that's on all the time, then this is ideal. However, even if it's a desktop that's only turned on periodically, this is still better than not having any protection.

First, you need to grab the software from the OSSEC website [1]. This will come as a tarball file called something like `ossec-hids-2.8.3.tar.gz`. Extract this to any directory; inside you should find a file called `install.sh`, which unsurprisingly is a shell script to install the software. You can run this by opening a terminal and navigating to the newly uncompressed directory and running:

```
sudo ./install.sh
```

This script is a little more involved than most software installs because it also creates a configuration file that sets OSSEC running the correct way. The first question you'll be asked after setting the language and confirming that you want to install is: What type of install do you want to create (server, agent, local, or hybrid)? The server option creates a master version of OSSEC that collates input from other machines into one place so you can see problems with any of your machines from a single point. Agent is for machines that monitor their local machine but send all the output to a server. Local only monitors the local machine, and hybrid is a combination of server and agent where it'll collate information from other machines, but also monitor the local machine. We'll use hybrid.

The rest of the options are quite self-explanatory, and you can just select the defaults (though if you don't have your email provider's servers' information, you may want to disable email alerts for now). Once OSSEC has all the details it needs, it'll compile the software (Figure 1).

Tripwire

OSSEC is a great all-round host-based IDS for one or more systems, but it's not the only option. Tripwire is perhaps the classic Linux HIDS and has been popular for many years for a good reason. It's rock solid and was the first bit of software to bring some intrusion detection features to the Linux world. There's a basic version of Tripwire that's open source and works well if you're protecting a single host. However, if you need more features – particularly if you're running on multiple machines – this version can be a little clunky to manage. There's a commercial version that offers many more features, but the cost of this makes it prohibitive for home use.

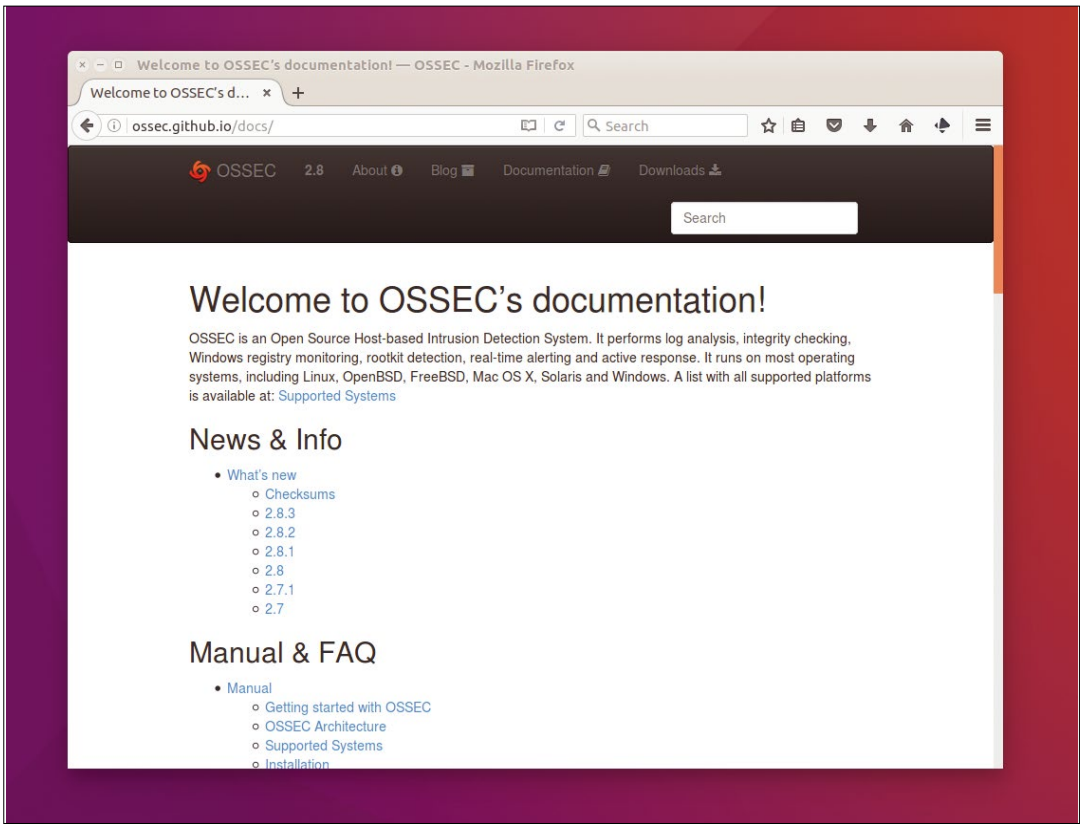


Figure 1: The OSSEC documentation on GitHub [2] provides plenty of details on how to customize OSSEC in more advanced ways than we have here.

After everything is set up, you'll see some output giving you directions on how to start and stop OSSEC, which will probably be `/var/ossec/bin/ossec_control start`. Run this (as root) to start monitoring your machine.

More Power

There are two ways of adding more machines to your protected system: As an agent or agentlessly. When a new machine is added agentlessly, you don't need to install anything on that machine. Instead, you configure the master to connect to the machine periodically and poke around inside to make sure everything's still as it should be. This is easier to set up and can be done even on machines that you can't put more software on. However, OSSEC can't tell as much about an agentless system, so you don't get the same level of protection. On the other hand, with an agent you have to install OSSEC, and it sends everything back.

Installing an agent on another machine is more or less the same process as installing the server. You'll need to grab the code and run `install.sh`, but this time select agent as the install type. Once it's installed, you need to add it to the server config. Obviously, security is critical to this process; otherwise, anyone who compromised your servers could simply fake the OSSEC agent to make the master think that everything was fine. OSSEC ensures security by generating keys on the master server that are then installed on the client. These are unique to each client and held securely on the agent machine.

The process of adding keys begins on the master server where you need to run the `manage_agents` program, which is installed in the OSSEC binary directory (typically `/var/ossec/bin`). Running this command-line program will give you the option to add a new agent, and then you'll need to enter the IP address or host name of the new machine. Once the new agent is added, you can extract the

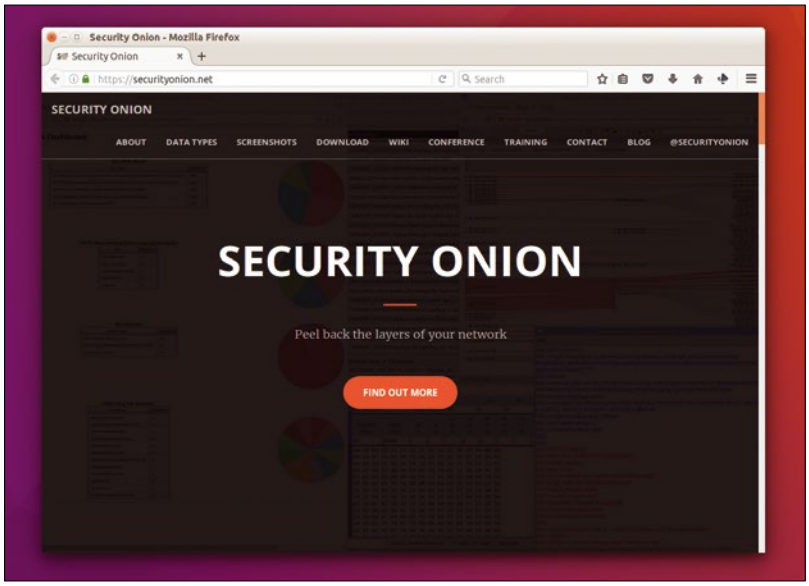


Figure 2: If you're serious about security, you might want to set up a machine just to monitor everything, and a specialist distro such as Security Onion comes with everything you'll need.

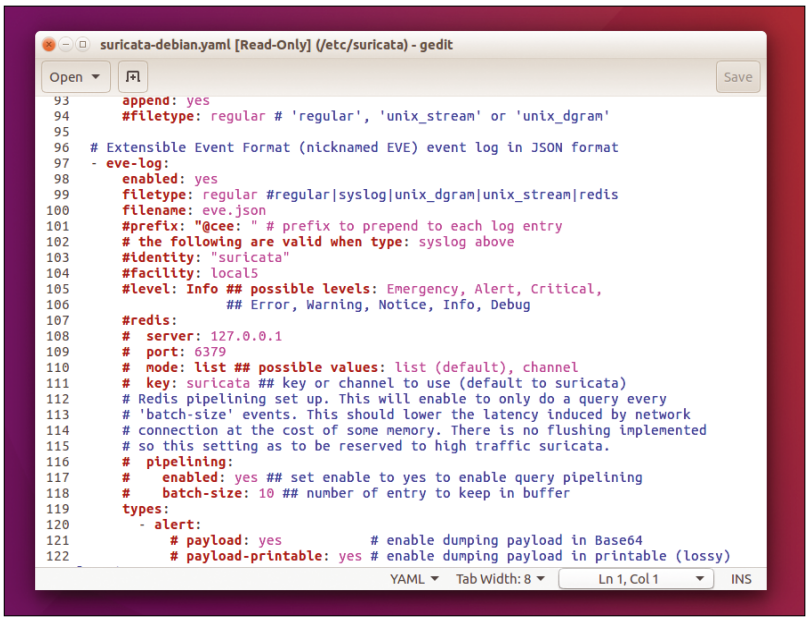


Figure 3: The default Suricata configuration file comes with plenty of comments if you want to tweak the way the software runs.

keys. This is done with the same program but by selecting E (extract) at the menu. You'll need to enter the agent's ID for which to extract keys. You should then see a string of what appears to be random text on the screen. You'll need this to set up the agent.

On the machine that you want to monitor, you also need to run the `manage_agents` program, but this time you'll be presented with a different menu. Select *Import Key From Server* and now paste in the key that you got in the previous step. Once this is added, you can restart OSSEC on both machines, and you should have communication between the agent and the server. Any actions on the agent will be reported in the logs on the master.

Double-No 7

Sometimes, you'll come across machines that you need to monitor but can't install software on – typically, these are embedded platforms such as routers. OSSEC can do some checking of these machines through the agentless configuration options. These options tell OSSEC to SSH into the machine and perform the commands that would run locally. OSSEC can read the results and report on any changes as though an agent were installed on that machine. However, OSSEC can't run as many security checks when running agentlessly as it can when it's installed on a machine, so this should only be used where running an agent isn't an option.

To begin, you'll need to enable agentless configuration with the following (run in the OSSEC binary directory):

```
sudo ./ossec-control enable agentless
```

Then, you need to add the machines you want to monitor with a command like the following (with user and host set correctly):

```
./register_host.sh add user@host
```

You'll need to set up either certificate or password SSH login for this to work.

Now you've got everything set up, the only thing left to do is check that nothing untoward is happening. If you set up email monitoring, you'll be contacted should anything happen. Otherwise, you'll need to keep an eye on the logs (typically stored in `/var/ossec/logs`). The `alerts.log` has all the critical information in it. Take a look at the "Logging" box for more details of how to make this easy.

This will get you up and running with OSSEC, and the default setup is quite good – certainly a lot better than running without any intrusion detection system (Figure 2). If you want to customize the system to your particular needs, you can do this by editing the `ossec.conf` file (typically in `/var/ossec/etc/`).

Looking at the Network

OSSEC monitors your files and machine, but there's another place that you can look to help identify any intruders – the network data. A NIDS monitors any data going in or out of the machine (and possibly other traffic it can detect on the network) to look for tell-tale signs of an attack. This is particularly good at spotting things like port scanning or trojan horses phoning home.

Snort

Snort is perhaps the most famous IDS for Linux and fundamentally works in a similar way to Suricata. Snort is incredibly powerful, but does take a little bit of effort to set up correctly. If you need maximum protection for an enterprise setup, then Snort is definitely worth investigating; however for smaller, less high-risk setups, Suricata should provide ample protection.

If you do decide to use Snort, some additional pieces of software can help you get up and running more easily. Pulled Pork is a command-line program to help manage your rules and configuration files. This is useful for Snort because NIDS (unlike most other software) need to regularly update their configurations to help identify newer trojan horses and hacking tools.

There is also a range of GUIs to help understand the Snort logs. Snorby, for example, is a Ruby On Rails web app that can convert the Snort logs into something easier to understand at a glance. Another option here is to use a more general logfile analysis tool.

Logging

You may have noticed that both the intrusion detection systems we've looked at here log activity in the same way that your system log does. This is intentional, and an important security step is keeping an eye on what's happening in your logs. Obviously, very few people find it interesting to read their logs on a daily basis, so if you've got more than one or two machines, it's a good idea to consolidate your logs into a central place and keep an eye on them in a more natural form than raw log files.

Logstash along with Elasticsearch and Kibana makes the ELK stack and is an excellent open source offering for larger groups of Linux machines (but is probably a little overkill for a small setup). Logwatch is another tool for consolidating log files into reports and is a bit easier to setup than ELK. For smaller setups, a simpler tool such as Glogg (Figure 4) can provide a more user-friendly log overview than a normal text viewer.

There are quite a few NIDS available, and we're going to look at one of the easiest to get up and running – Suricata. (See the "Snort" box for other options.)

There are two basic steps to installing Suricata. First, you need the software which should be in your distro's repository. On Ubuntu-based systems, you can get it with:

```
sudo apt install suricata
```

The second thing you need are the rules. These are the configuration files that tells Suricata how to identify suspicious data on the network. There are loads of different rule sources, but a good source is Emerging Threats. You can download the `emerging.rules.tar.gz` file [3], and then extract it and copy the rules into `/etc/suricata/rules`.

You'll also need a configuration file (Figure 3). On Ubuntu systems, the default is `/etc/suricata/suricata-debian.yaml`, but on many distributions it's `/etc/suricata/suricata.yaml`. This default configuration file should be fine for testing out the software, so let's jump in and start running with:

```
sudo suricata -c /etc/suricata/
  suricata-debian.yaml -i <iface>
```

where `<iface>` is the ID of the network interface you want to monitor. If you don't know what this is, enter `ip addr` into a terminal to see a list of all options. For example, on our test system, we see the following line in the output:

```
2: enp3s0: <BROADCAST,MULTICAST,
  UP,LOWER_UP> mtu 1500 qdisc
  pfifo_fast state UP group default qlen 1000
```

The interface name for this network connection is enp3s0.

If all goes well, you should see some output saying that Suricata is starting. If there are any errors,

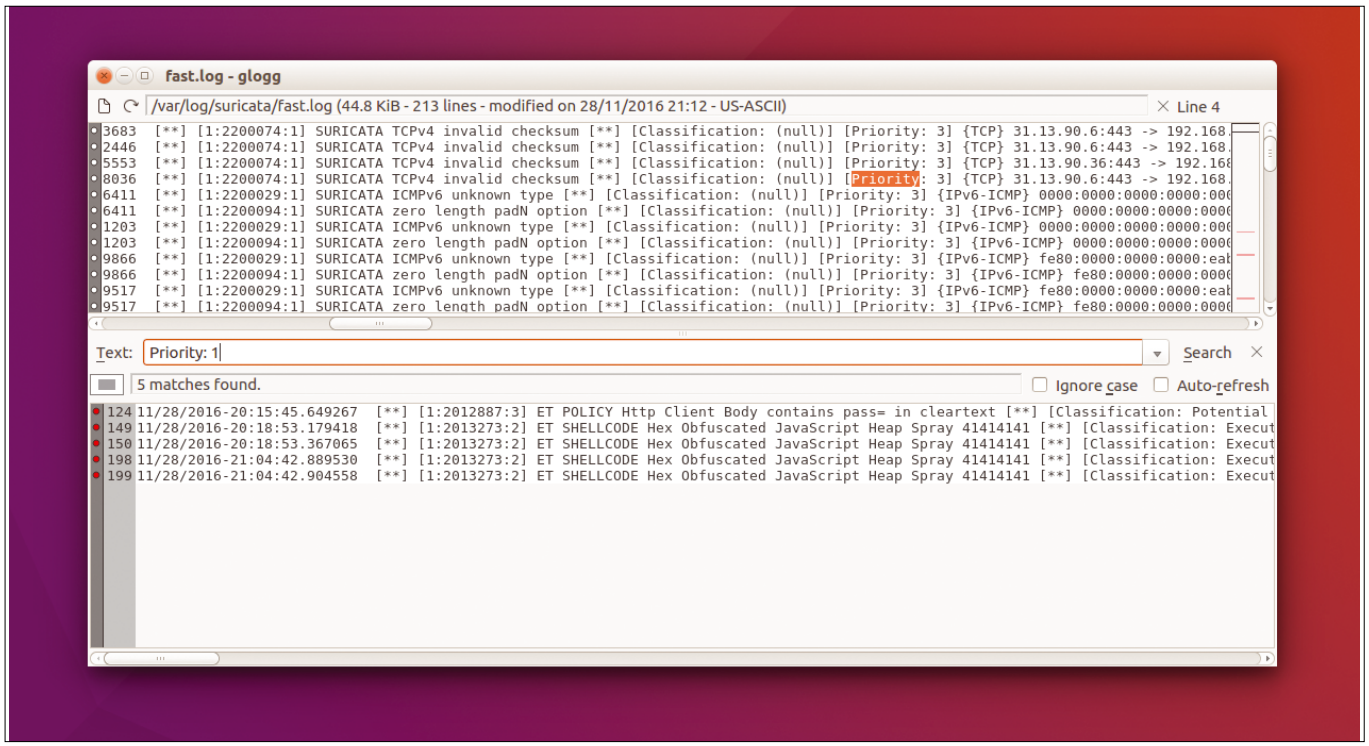


Figure 4: Using Glogg, we can easily monitor high-priority issues. Grep on the command line can do the same thing.

you'll need to fix them before continuing. Once it's started correctly, enter Ctrl+C to end the process and return to the shell. We'll now set up the service to run automatically.

Before running Suricata as a service, you'll need to make sure the defaults are set correctly. These are in the file `/etc/defaults/suricata`. Most importantly, make sure that the following line is present:

```
RUN=yes
```

Without this, the service will refuse to run. The second thing you need to check is the listening mode. NFQueue is often set as the default even though this isn't available on all kernels. The pcap mode should work on more setups, but does require you to set the interface to listen on. On our test system, we configured Suricata with the following.

```
LISTENMODE=pcap
IFACE=enp3s0
```

The last thing we need to do before kicking off the service is to change the Suricata logging. When we started Suricata from the command line, we saw the output in the terminal, but obviously this won't work when we start as a service, so we need to change the main YAML configuration file to output data to the correct place.

Open the configuration file (in Ubuntu this is `/etc/suricata/suricata-debian.yaml`, but on other distributions it could be `/etc/suricata/suricata.yaml`) and make sure the logging outputs section is changed to the following:

```
logging:
  outputs:
  - console:
      enabled: no
      # type: json
  - file:
      enabled: yes
      filename: /var/log/suricata/suricata.log
```

With this set up, you can start Suricata with:

```
service suricata start.
```

Now we've got our Suricata intrusion detection service (IDS) running, let's take a look at the logs. These are typically in `/var/log/suricata`. The `fast.log` gives an overview of any security-critical events whereas `http.log` will give details of all HTTP activity, and `stats.log` will – unsurprisingly – contain statistics. If you enable email reports in the YAML configuration file, then you'll have all important information sent to you via email; otherwise, you'll have to keep an eye on the logs (see the "Logging" box for more).

With everything set up, you're now monitoring for any changes in the key files on all machines, looking for any changes to configuration of the security services (e.g., iptables) and monitoring network traffic for any suspicious activity. If you want to go one step further, you can actively try to trap attackers (see the "Honeypots" box for details). Remember, though, that no security is impervious to attack; it only makes it more difficult. A sufficiently skilled or determined attacker may still be able to infiltrate your machines, and no security setup is sufficient to stop thinking about digital self-defense. However, with these IDS in place, you've got another two layers of protection. ■■■

Honeypots

Intrusion detection systems try to separate out normal use from malicious use, but this is a difficult task. False positives and missed attacks can happen and indeed are inevitable on any IDS. Another approach is to have a computer (possibly a cheap one such as a Raspberry Pi) that shouldn't have any activity on it. Anything that's run on it is then automatically cause for alarm. This is known as a honeypot, because it's trying to attract attackers into a trap. This could be a normal system that you set to listen for connection attempts, but you can make it more attractive by using honeypot software that mimics insecure software and then records the attacker's action.

The Kippo SSH honeypot is one of the most popular options. To an attacker it appears just like an SSH server. You can set it to accept common passwords, and it'll let the attacker log into what appears to be an SSH session. However, they're locked away from any tools that can do any damage, and all their actions are recorded. When linked to a public IP address, honeypots give you an idea of just how many attackers there are searching the web for any weak points. When added to your internal network, they are another tool to help detect any attackers that have breached your defenses.

Info

[1] OSSEC: *http://ossec.github.io/downloads.html*

[2] OSSEC documentation: *ossec.github.io/docs/*

[3] Rules file: *http://rules.emergingthreats.net/open/suricata-1.3/*

Set up your own cloud
Get the most from Nextcloud

Get all the benefits of cloud storage and calendars without governments and megacorps spying on you.

BY MIKE SAUNDERS

Maybe you've heard this line before: "There is no cloud, just other people's computers." Indeed, if you've been to a Free Software-related event recently, you may even have a sticker with those words on it, courtesy of the Free Software Foundation Europe [1]. And it's a good point: "cloud computing," despite all the hype, is simply about handing over control of your data and resources to someone else. It's not actually especially new, but a bunch of technologies mean you can effectively "live" in the cloud, using a low-spec computer and doing all your processing and storage elsewhere. (Well, providing you always have a good Internet connection, of course.)

Still, there are some benefits to the cloud. If you're regularly switching between lots of devices and machines, having easy access to your data is a major bonus. Similarly, you can use cloud services to make regular backups of your data with minimal intervention. So what do you do if you like the technical benefits that cloud services bring, but don't want to hand over all your data to Google, Dropbox, or anyone else? Make your own cloud, of course!

Now, this may all sound extremely complicated when you consider all the things available in typical cloud services: file storage, file sharing, calendars, document collaboration, and more. Fortunately, though, there's an excellent Free Software solution in the form of Nextcloud. This is a spin-off of ownCloud (see the "ownCloud vs. Nextcloud" box for more information on its origins) and really simplifies the job of setting up your own cloud infrastructure. You can set up Nextcloud in your home or office for a "local" cloud installation – or for something more Google or Dropbox-esque, host it on a remote server so you can access your data, calendar, and documents everywhere. Ultimately, like all good things in free and open source software (FOSS), the control remains with you.

In this article, I will show you how to install, configure, and use Nextcloud. It's not a hugely difficult job, but there are various steps involved, including editing some configuration files. If you're familiar with command-line basics, you should be able to follow. Let's get started.

Setting up the Server

First, you'll need to grab the latest Nextcloud release from the website [2]. Under "Get Nextcloud Server," click the *Download* button, which will pop up a box offering the most recent release – 10.0.1 at the time of writing. Click the large blue *Download* button and save the file to your home directory; it's around 45MB.

Next, you need to install some dependencies for Nextcloud. The core requirements are a web server (Apache) along with a database (MySQL or MariaDB) and various PHP libraries. To get everything you need on a Debian-based distro (I'm using Debian 8 here), enter the following as root:

```
apt-get install apache2 libapache2-mod-php5 ➥
  php5-sqlite php5-gd php-xml-parser php5-intl ➥
  php5-mcrypt php5-curl php5-imagick php-apc ➥
  php5-mysql mysql-server ntp
```

If you're on a different distro, search in your package manager for Apache, MySQL, and those PHP modules – they may have similar names, or you may have to check your distro's documentation. In any case, once you have everything installed, Apache and MySQL should start automatically; if not, try the following as root:

```
service apache2 start
service mysql start
```

(On systemd distros, try changing `service` to `systemctl` if the above doesn't work.) You'll then be able to check whether Apache is running by going to the IP address or server's hostname in a web browser – or on the same machine, go to `http://localhost`. You should see a default page for Apache, confirming that the web server is running.

Next up is copying the Nextcloud files into your web server's directory. On Debian and Ubuntu-

based distros, this can be found in `/var/www/html` – but check with your distro's setup in case it's somewhere else. On Debian, you can extract the downloaded `.zip` file as root like so:

```
cd /var/www/html
unzip /home/mike/nextcloud-10.0.1.zip
```

(Change the location to where you downloaded Nextcloud accordingly.) This creates a new directory, `/var/www/html/nextcloud`, with all the files for the server. You're almost ready to start Nextcloud's web-based setup wizard, but there are a few Apache tweaks you need to do beforehand. First, change the Nextcloud installation directory's permissions so that the Apache user can modify them:

```
chown -R www-data:www-data /var/www/html/ ➥
  nextcloud
```

Again, `www-data` is what Apache runs under in Debian, but it may be different in your specific distro (it's `apache` in Fedora and `http` in Arch Linux). You also need to give Apache some information about the Nextcloud installation, so create `/etc/apache2/sites-available/nextcloud.conf` as shown in Listing 1.

Modify the directory paths accordingly for your Nextcloud installation. Enable this configuration by creating a symbolic link to it like so:

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

Finally, enable some useful modules in Apache with the following commands:

```
a2enmod rewrite headers env dir mime
```

and restart Apache:

```
service apache2 restart
```

All being well, Apache is now fully set up for Nextcloud, so you can put the command line aside for a while and switch to the shiny web-based setup wizard.

Installation Wizard

In a web browser, go to the hostname of the machine running Apache and add `/nextcloud` after it. For instance, if your web browser is running on the same machine as Apache, you can just visit `http://localhost/nextcloud`. The setup wizard should appear, prompting you to enter an administrator username and password (Figure 1). Don't forget these! They're like Nextcloud's root account and are essential to perform most configuration jobs.

Underneath the admin box, you'll see a "Performance warning" message. By default, Nextcloud is set up to use SQLite for storage – which is fine for testing, but rather slow for production use. Click *Storage & database* and choose *MySQL/MariaDB* underneath. Then enter the root or admin user for your MySQL/MariaDB installation and password; if you're on Debian, you will have created those when you apt-getted the database earlier. Finally, provide a name for the database (e.g., "mynextcloud") and click *Finish setup*.

If all is well, you'll then land at the main Nextcloud file sharing screen (Figure 2). (A pop-up may appear with links to downloading Nextcloud clients – just dismiss it for now.) And, there you have it, you're ready to use Nextcloud! Well, as the admin user, at least. Obviously, this is not good practice on a day-to-day basis, so your first job is to click the *admin* drop-down list in the top-right of the interface, and then *Users*. Here you can enter new usernames and passwords in the empty fields and then click *Create*.

By default, Nextcloud is set up primarily for file sharing, but you may want to enable much more of its functionality. Click the

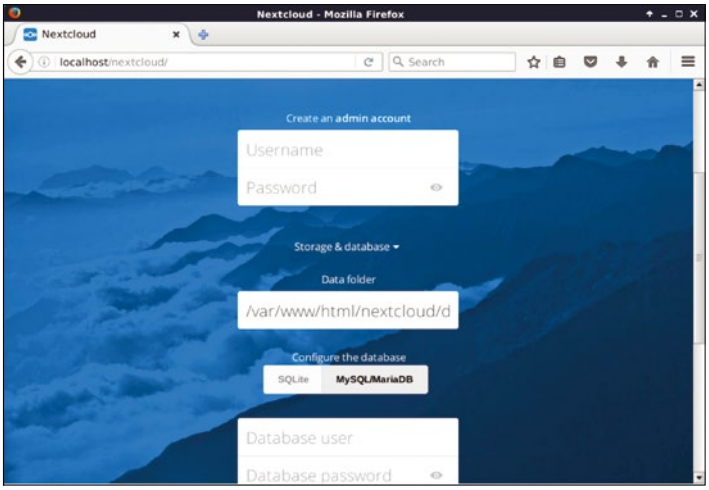


Figure 1: Nextcloud needs a bit of command-line preparation, but most of the setup is achieved through the web interface.

Listing 1: Nextcloud Installation Information

```
Alias /nextcloud "/var/www/html/nextcloud/"

<Directory /var/www/html/nextcloud/>
  Options +FollowSymlinks
  AllowOverride All

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

 SetEnv HOME /var/www/html/nextcloud
 SetEnv HTTP_HOME /var/www/html/nextcloud

</Directory>
```

drop-down menu in the top-left of the web interface, next to the Nextcloud logo, and choose *Apps*. This is where you can enable various add-ons, so browse around and see what takes your fancy. I especially recommend Calendar, Contacts, and Documents in the Productivity section, bringing Nextcloud much closer to the services offered by Google (Figure 3).

Using Nextcloud and Advanced Tuning

Now you're ready to use Nextcloud for your day-to-day work. Click *admin* in the top right and then *Log out*. Now log in as the normal user account you created earlier, and you'll arrive at the file management page. You can now begin using the file storage features: create new folders and upload your files knowing that they're backed up on your own personal cloud. To share a file or folder, click the *Share* button (two lines connected with dots), which lets you share with other users on the same Nextcloud installation, or indeed pass around a link over the web (be careful with this). By clicking the three separate dots next to a file or folder, you can download, rename, or delete it.

Click *Files* in the top-left to access Nextcloud's other features; try Calendar first. This brings up a familiar calendar interface, where you can navigate between months and years, click on days to add events, and so forth. You may want to access your calendar from other software as well: click the three dots next to the calendar name (e.g., "Personal") and then *Link*. This provides a URL that you can then use in WebDAV-compatible calendars.

Also in the top-left menu, try Documents, and click the `About.odt` file (Figure 4). As the filename extension suggests, Nextcloud uses the Open Document Format (like LibreOffice), although its editor is rather primitive in comparison. Still, it's more than good enough for basic editing tasks, and multiple users can work on the same document simultaneously (click the *Share* button at the top).

Depending on your desktop environment, you may be able to access your Nextcloud files via you regular file manager (Figure 5), instead of using the web interface. In Nextcloud, click the top-left menu and choose *Files*. Then, look for *Settings* in the bottom-left and click it – a *WebDAV* box will appear. This is an address you can use in your file manager to access the Nextcloud data via the WebDAV protocol.

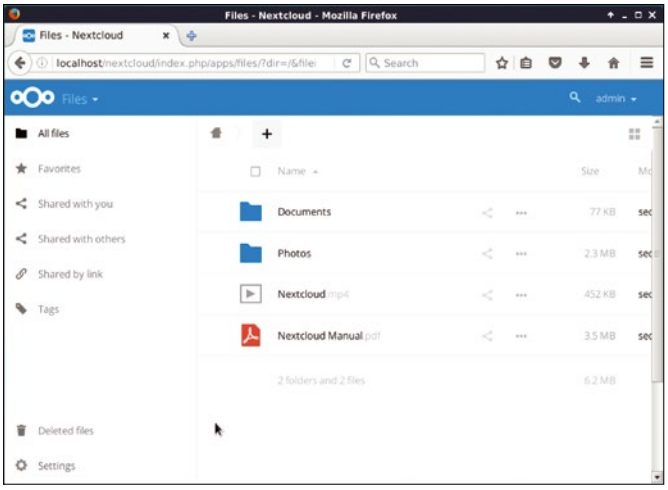


Figure 2: Nextcloud's file manager is a simple affair, but gets the job done nicely.

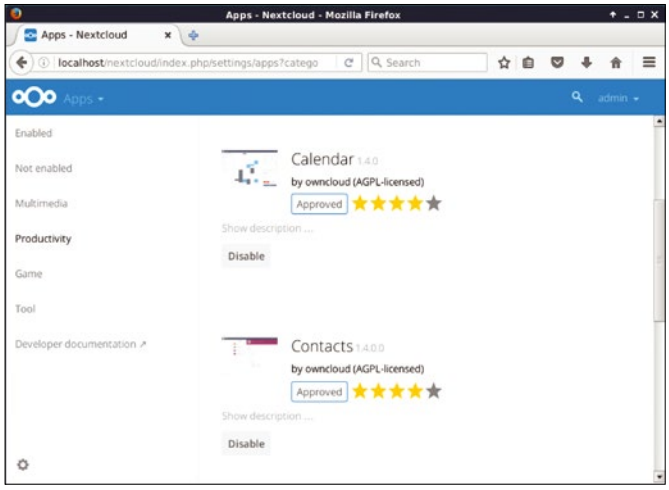


Figure 3: Beef up your Nextcloud with extra multimedia tools, productivity apps, and even games.

ownCloud vs. Nextcloud

Forks in the Free Software world are rarely pleasant but often required in order for progress to be achieved. Look at X.org which forked off from XFree86, for instance, or LibreOffice which forked off from OpenOffice. Both of those forks ended up attracting by far the most developers and were adopted by virtually every Linux distro available. A similar situation exists with ownCloud and Nextcloud.

The ownCloud project was originally created by KDE developer Frank Karlitschek in 2010. One year later, Karlitschek started a company to monetize ownCloud by offering an enterprise version on top of the open source community one. Things seemed to go well for ownCloud until April 2016, when Karlitschek and some other developers left the company and started a fork: Nextcloud.

Karlitschek hasn't elaborated on exactly why he made the fork, but said that some things "could have been better at ownCloud Inc." Many followers speculate that the company Karlitschek founded was losing its community focus as more investors got involved, hence the need for a fork to return to its roots.

In any case, it looks like Nextcloud will be the more popular and developed version for the foreseeable future – at least for FOSS fans. ownCloud has more brand recognition after six years of development, but it remains to be seen whether it can attract new developers, or whether enough damage has been done.

Copy it to your clipboard and then try to open a new location in your file manager; the specifics will depend on your desktop. In Nautilus, for instance, hitting Ctrl+L lets you paste in an address. I had to change `http://` to `dav://` in my instance, but your results may vary depending on the libraries you have installed. If in doubt, check your desktop's documentation.

Nextcloud periodically needs to perform some background tasks to keep it ticking over smoothly. By default, Nextcloud does this whenever a user accesses the web interface, but that's far from ideal. A better solution is to set up a cron job that runs every 15 minutes. Edit the cron table by entering the following:

```
crontab -u www-data -e
```

(Change `www-data` for the appropriate account for the web server.) Then add this to the bottom:

```
*/15  *  *  *  * php -f /var/www/html/ ➥
  nextcloud/cron.php
```

Back in the Nextcloud web interface, log in as the admin user, click the top-right menu and go to Admin. You'll see a cron section there where you can switch from Ajax to cron for better performance.

In your Nextcloud installation directory, check out `config/config.php`. This contains some useful information, such as the username and password that Nextcloud created for its MySQL/MariaDB database – in case you need to perform some manual work. Also worth noting here is the `trusted_domains` option, which is where you can specify IP addresses and hostnames that are valid for this Nextcloud installation. You can see that there's already one item (numbered zero) in the list for `localhost`, so add more if you want to access it under a different name (e.g., if the hostname on the network is `raspberrypi`, add that here).

By default, Nextcloud uses plain HTTP, which works for a simple home network or testing, but if you want to use it in business or over the Internet, you should enable SSL. The Nextcloud installation guide [3] explains how to do this and includes many other pointers for advanced configuration and fine-tuning of your Nextcloud installation. It's well worth checking out.

So, enjoy your shiny new self-hosted cloud, as well as independence from snooping eyes. Let us know how you get on, and what funky skillo things you're doing with your Nextcloud installation! Indeed, you could try hosting Nextcloud on a spare Raspberry Pi (see the box "Dig out that Raspberry Pi!" for details). ■■■

Dig out that Raspberry Pi!

Got a spare Raspberry Pi sitting around doing nothing especially useful? Put it to work making backups of your critical data! Yes, Nextcloud will run on the Rasp Pi, but don't expect blazing performance, even on the Raspberry Pi 3. If your Nextcloud installation will be processing vast amounts of data and documents for many different users, the Rasp Pi may be too sluggish – but it's more than adequate for occasional jobs.

I, for instance, have a Rasp Pi plugged in to my router via Ethernet, running Nextcloud from a 16GB SD card. That may not sound like much data, and it's not used for storing videos or music; however, for documents and other work-related files, it's more than spacious enough. With the NextCloud desktop client installed and a link to its shared folder on the desktop, anything saved in there is automatically synced and stored on the Rasp Pi as well – without having to do any extra work.

With all critical data backed up on the Pi, it's not the end of the world if a particular desktop or laptop goes kaput. Switch to a different one, sync up with the Rasp Pi, and all files are accessible via a desktop folder again. Even when testing other operating systems inside a virtual machine, for which no Nextcloud desktop clients are available, the web interface is still there as an option.

Info

[1] "There is no cloud" sticker: *https://fsfe.org/contribute/spreadtheword.en.html#nocloud*

[2] Download Nextcloud: *https://nextcloud.com/install/*

[3] Nextcloud installation guide: *http://tinyurl.com/hgpqhe8*

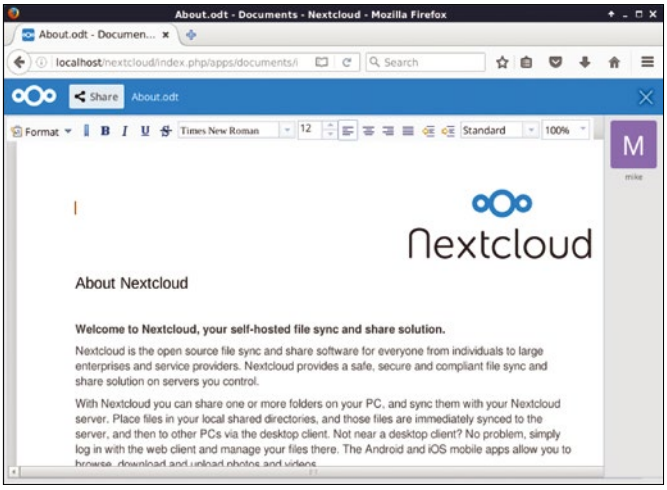


Figure 4: It's no LibreOffice, but the collaborative document editor works well for basic tasks.

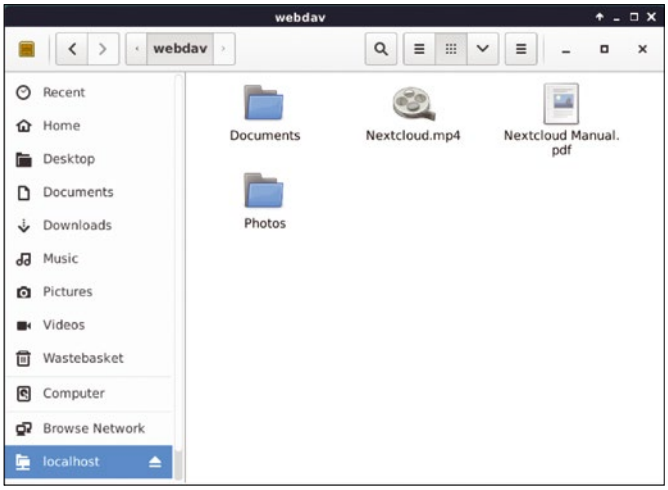


Figure 5: Desktop integration in action – here's Nautilus accessing Nextcloud's files over WebDAV.

FEATURED EVENTS

Users, developers, and vendors meet at Linux events around the world. We at *Linux Magazine* are proud to sponsor the Featured Events shown here.

For other events near you, check our extensive events calendar online at *http://linux-magazine.com/events*.

If you know of another Linux event you would like us to add to our calendar, please send a message with all the details to *events@linux-magazine.com*.

SCaLE 15x

Date: March 2–5, 2017

Location: Pasadena, California

Website: *https://www.socallinuxexpo.org/scale/15x/*

Don't miss the 15th annual Southern California Linux Expo at the Pasadena Convention Center. SCaLE 15x is four days packed with more than 100 technical sessions and featuring more than 100 exhibitors. More than 3,600 Linux and technology professionals attended SCaLE 14x in 2016. Visit the website to learn more about the event.

2017 HPC for Wall Street

Date: April 3, 2017

Location: New York City, New York

Website: *http://www.flaggmgmt.com/linux/*

Register now for HPC for Wall Street for an All-Star Conference Program! This event is designed for forward-thinking industry veterans to get the latest on cloud and data centers. Visit the website to learn more and register today.

DrupalCon Baltimore

Date: April 24–28, 2017

Location: Baltimore, Maryland

Website: *https://events.drupal.org/baltimore2017*

The Drupal community is one of the largest open source communities in the world. We're developers, designers, strategists, coordinators, editors, translators, and more. Once a year, our community comes together in a US city for one of our biggest events: DrupalCon. This year we are excited to bring DrupalCon to Baltimore. Visit our website to learn more!

EVENTS

Univention Summit	January 26	Bremen, Germany	https://www.univention-summit.de/
DEVCONF.cz	January 27–29	Brno, Czech Republic	https://devconf.cz/
DevConf Panamá 2017	February 1–3	Panama City, Panama	http://devconfpanama.com/#/
FOSDEM 2017	February 4–5	Brussels, Belgium	https://fosdem.org/
Open Source Leadership Summit	February 14–16	Lake Tahoe, California	http://events.linuxfoundation.org/events/open-source-leadership-summit
Embedded Linux Conference	February 21–23	Portland, Oregon	http://events.linuxfoundation.org/events/embedded-linux-conference
OpenIoT Summit	February 21–23	Portland, Oregon	http://events.linuxfoundation.org/events/openiot-summit
SCaLE 15x	March 2–5	Pasadena, California	https://www.socallinuxexpo.org/scale/15x/
CeBIT 2017	March 20–24	Hanover, Germany	http://www.cebit.de/home
SPTechCon 2017	April 2–5	Austin, Texas	http://www.sptechcon.com/
2017 HPC for Wall Street – Cloud and Data Centers Show and Conference	April 3	New York, New York	http://www.flaggmgmt.com/linux/
JAX DevOps	April 3–6	London, United Kingdom	https://devops.jaxlondon.com/
DrupalCon Baltimore	April 24–28	Baltimore, Maryland	https://events.drupal.org/baltimore2017
Check_MK Conference #3	May 2–4	Munich, Germany	http://mathias-kettner.de/
ISC High Performance (ISC 2017)	June 18–22	Frankfurt, Germany	http://www.isc-hpc.com/
AnDevCon	July 17–19	Washington, DC	http://www.andevcon.com/
InterDrone	September 6–8	Las Vegas, Nevada	http://www.interdrone.com/

CALL FOR PAPERS

We are always looking for good articles on Linux and the tools of the Linux environment. Although we will consider any topic, the following themes are of special interest:

- System administration
- Useful tips and tools
- Security, both news and techniques
- Product reviews, especially from real-world experience
- Community news and projects

If you have an idea, send a proposal with an outline, an estimate of the length, a description of your background, and contact information to *edit@linux-magazine.com*.

The technical level of the article should be consistent with what you normally read in *Linux Magazine*. Remember that *Linux Magazine* is read in many countries, and your article may be translated into one of our sister publications. Therefore, it is best to avoid using slang and idioms that might not be understood by all readers.

Be careful when referring to dates or events in the future. Many weeks could pass between your manuscript submission and the final copy reaching the reader's hands. When submitting proposals or manuscripts, please use a subject line in your email message that helps us identify your message as an article proposal. Screenshots and other supporting materials are always welcome.

Additional information is available at:
http://www.linux-magazine.com/contact/write_for_us.

AUTHORS

CONTACT INFO

Editor in Chief
Joe Casad, jcasad@linux-magazine.com

Managing Editor
Rita L Sooby, rsooby@linux-magazine.com

Localization & Translation
Ian Travis

News Editor
Swapnil Bhartiya

Copy Editors
Amber Ankerholz, Amy Pettle

Layout
Dena Friesen, Lori White

Cover Design
Dena Friesen

Cover Image
© Lukas Gojda, 123RF.com

Advertising – North America
Ann Jesse, ajesse@linuxnewmedia.com
phone +1 785 841 8834

Advertising – Europe
Brian Osborn, bosborn@linuxnewmedia.com
phone +49 89 99 34 11 48

Publisher
Brian Osborn, bosborn@linuxnewmedia.com

Marketing Communications
Gwen Clark, gclark@linuxnewmedia.com
Linux New Media USA, LLC
616 Kentucky St.
Lawrence, KS 66044 USA

Customer Service / Subscription
For USA and Canada:
Email: cs@linuxpromagazine.com
Phone: 1-866-247-2802
(Toll Free from the US and Canada)
Fax: 1-785-856-3084

For all other countries:
Email: subs@linux-magazine.com
Phone: +49 89 99 34 11 67
Fax: +49 89 99 34 11 98

www.linuxpromagazine.com – North America

www.linux-magazine.com – Worldwide

Printed in Germany

Distributed by COMAG Specialist, Tavistock Road, West Drayton, Middlesex, UB7 7QE, United Kingdom

LINUX PRO MAGAZINE (ISSN 1752-9050) is published monthly by Linux New Media USA, LLC, 616 Kentucky St., Lawrence, KS, 66044, USA. Periodicals Postage paid at Lawrence, KS and additional mailing offices. Ride-Along Enclosed. POSTMASTER: Please send address changes to Linux Pro Magazine, 616 Kentucky St., Lawrence, KS 66044, USA.

Published monthly in Europe as Linux Magazine (ISSN 1471-5678) by: Sparkhaus Media GmbH, Putzbrunner Str. 71, 81739 Munich, Germany.

Approximate On Sale Date

UK / Europe	Feb 04
USA / Canada	Mar 03
Australia	Apr 06

Issue 196 / March 2017

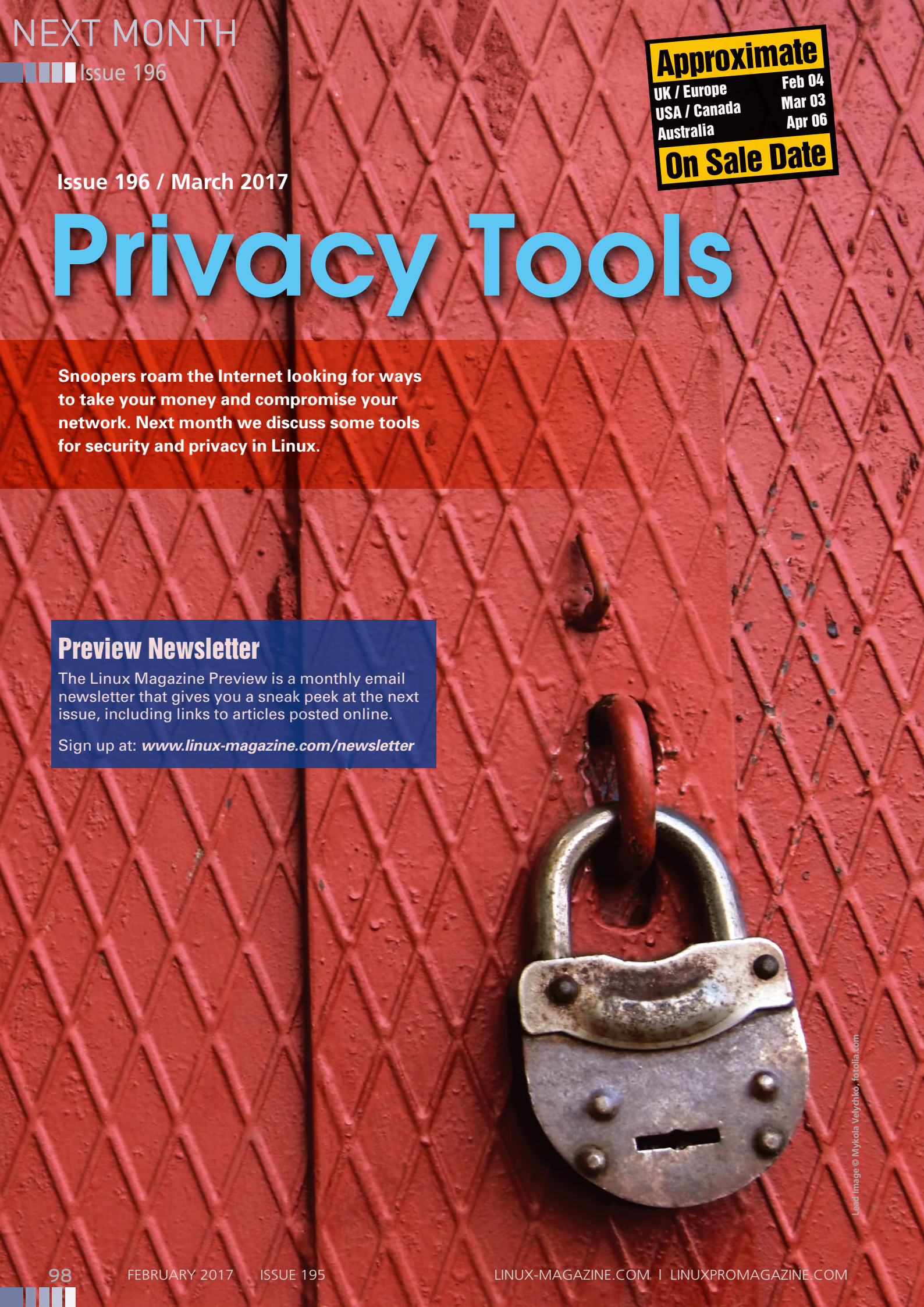
Privacy Tools

Snoopers roam the Internet looking for ways to take your money and compromise your network. Next month we discuss some tools for security and privacy in Linux.

Preview Newsletter

The Linux Magazine Preview is a monthly email newsletter that gives you a sneak peek at the next issue, including links to articles posted online.

Sign up at: ***www.linux-magazine.com/newsletter***



Lead Image © Mykola Velychko, fotolia.com

DrupalCon Baltimore

The Drupal community is one of the largest open source communities in the world. We're developers, designers, strategists, coordinators, editors, translators, and more. Each year, we meet at DrupalCamps, meetups, and other events in more than 200 countries. But once a year, our community comes together in a U.S. city for our largest gathering, **DrupalCon**. This year we're going to Baltimore!

Tickets on sale now

https://events.drupal.org/baltimore2017

Important Dates

Call for Papers closes: February 1

Early Bird ticket pricing ends: March 3

Schedule published: March 15

MicroBlade

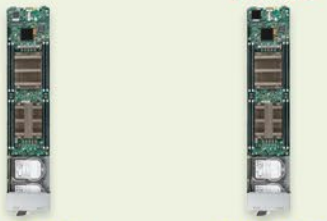
High Density • High Performance • High Efficiency • Cost-Effective
Enterprise, Data Center, Web Applications, HPC and Cloud Computing Solutions

Intel® Xeon® Processor E5-2600 v4/v3 Product Families Supported

3U MicroBlade
*Up to **28 UP/ 14 DP** Nodes*

6U MicroBlade
*Up to **112 UP / 28 DP** Nodes*

10Gbps



MBI-6128R-T2 MBI-6128R-T2X

Dual Intel® Xeon® Processor E5-2600 v4/v3

New! ***HD Graphics*** New!
NVMe
10Gbps

MBI-6219G-T7LX MBI-6119G-T7LX

Intel® Xeon® Processor E3-1578L v5

New! ***10Gbps*** New!

MBI-6218G-T81X
MBI-6218G-T41X
MBI-6118G-T81X
MBI-6118G-T41X

Intel® Xeon® Processor D

New! ***SAS3*** New! New! New!

MBI-6119G-C2 MBI-6119G-C4 MBI-6119G-T4 MBI-6219G-T

Intel® Xeon® Processor E3-1200 v5

MBI-6118D-T2/T2H MBI-6118D-T4/T4H

Intel® Xeon® Processor E3-1200 v4/v3

MBI-6418A-T7H MBI-6418A-T5H

Intel® Atom™ Processor

Intel Inside®.
Powerful Productivity
Outside.

TITANIUM We Power the Cloud X11 SERVER PLATFORMS nvm EXPRESS

Learn more at www.supermicro.com