CLUSTERED SAMBA
Distributed storage the easy way

CLUSTERED SAMBA

Distributed storage the easy way

NFS
Is this classic filesystem
ready for today's network?

Scrapy
Build your own web crawler

Collabora Suite
Get productive with
LibreOffice in the cloud

!@#%&^!
Charly catches his mistakes
using a cool tool with a name
we can't print on this cover

Gimp Tricks
Add out-of-bound effects
to your slide show

UNITY AND PURPOSE

Dear Linux Pro Reader,

It is 2010, and Ubuntu is riding the crest of a wave. The distro named for a principle of African philosophy that is "too beautiful to even say in English" had legions of loyal users and, perhaps more importantly, legions of enthusiastic volunteers, even though the project was actually backed by a for-profit company called Canonical.

Just when everything was going well, Ubuntu founder Mark Shuttleworth announced the distro would be giving up its specially themed Gnome desktop and switching to a brand new desktop they were going to call Unity.

First reactions were not good – people wanted their familiar Ubuntu Gnome desktop back. They said Unity was confusing and counterintuitive. Canonical said it was working on a plan to merge PCs with mobile phones, something they called "convergence," and Unity was part of that plan. But many in the community liked Ubuntu the way it was – why did it have to change just so Canonical could pursue its corporate business strategy?

Things got worse when it was revealed that Unity had the ability to beam user search results back to Canonical, Amazon, and other vendors. Richard Stallman of the Free Software Foundation roundly denounced Unity and Canonical for adding such a feature, and lots of rancor ensued.

Unity brought the end to a honeymoon for Ubuntu and its starry-eyed community, but no honeymoon lasts forever. Canonical really is a company, and it always was, and the point of business is to do business. Ubuntu's controversial search feature was officially disabled by default in version 16.04, but by then, Canonical was on to other things anyway, working on its OpenStack strategy and the large corporate contracts that will one day determine whether the company will survive.

As I look back on that era, perhaps the most interesting thing is that the Unity desktop is still surviving and, apparently, thriving. It is hard to get numbers on which desktops users are actually using, but a quick glance at the DistroWatch list of Hits Per Day as a measure of attention the distro is receiving from the Linux community shows that mainline Ubuntu with Unity is far outperforming the other officially sanctioned Ubuntu flavors, such as Kubuntu, Lubuntu, and Xubuntu. Even Ubuntu Mate, which was created just to give users a taste of the old Gnome 2 look and feel, does not get nearly as much attention as the mainline Unity version.

Note: Linux Mint, which leads the DistroWatch list, is also an Ubuntu derivative, although it is a separate operation that is not part of the Ubuntu project. Because Mint puts all its desktop versions in one place, rather than managing them through separate projects, it is impossible to say how much of the Mint attention is going to which desktop, but it is safe to conclude that probably no one is using Mint with Unity.

For better or worse, it appears that Unity is here to stay, which is quite surprising given the controversies that once surrounded it. Say what you want about the design or the potential for data mining (of which I am no fan), the saga of Unity speaks to Canonical's persistence and their continued focus on a long-term vision.

Possible reasons for the continued acceptance of Unity within the community despite the early problems are:

- Unity got better – it isn't as annoying as it used to be, because the Ubuntu team is actually pretty good at what they do, and they have steadily improved Unity based on user feedback.
- Enough new users who like Unity are replacing the old users who don't like it.
- The desktop doesn't matter – the underlying tools are pretty darn similar, so who cares which boxes and icons you click on. (I must confess I might be in this camp. Although I certainly agree that the system shouldn't spy on the user, in terms of ergonomic functionality, I have never really gotten too worked up over the Gnome vs. KDE, Gnome 2 vs. Gnome 3, or any of the other desktop dust-ups through the years.)
- People are downloading Ubuntu and then putting whatever desktop they want on the system afterward. This possibility is impossible to analyze based on the DistroWatch statistics, but if it were true, the fact that people would rather download Ubuntu and put LXDE on it than download Lubuntu, which includes LXDE by default, would have interesting implications as a study in branding and product definition.

What does it all mean? I'm certainly not going to be able to sort it all out in 800 words, but if you were around to see all the fallout in 2010 and 2011, it is worth pausing to note that Unity is still here.

Joe Casad,
Editor in Chief

LINUX PRO MAGAZINE

Clustered Samba

This month's issue studies file storage with two topics near and dear to network administrators' hearts: Samba and NFS.

REVIEWS

Community Notebook

HIGHLIGHTS

FEATURES

LINUXUSER

ISSUE 191 AUG 2016

ISSUE 191 AUG 2016

DVD ROM

TWO TERRIFIC DISTROS

DOUBLE-SIDED DVD!

On the DVD

Remix OS for PC (64-bit)

Remix OS is a PC-optimized version of a native Android OS that comes out of the Android-x86 open source project [1]. The distro prides itself on its speed, and the developers claim it will run on any computer and breathe new life into old hardware. The Remix folder on the DVD contains a TXT, ISO, and EXE file [2]. If you want to dual-boot Remix on an Ubuntu installation, a YouTube video will help you get started [3], along with the instructions on the XDA Developers site [4]. To set Remix up in Virtual-Box, just choose *Linux* as the Type and *Other Linux (64-bit)* as the Version. Allow Remix to use about half of your RAM, and give the virtual hard disk about 16GB. A YouTube video can help you with this setup, as well [5]. If you have a Windows box and want to dual boot Remix OS, the developers supply an EXE file, which you can use along with a video on the Remix OS website [6].

Gentoo Linux (Live)

Gentoo Linux [7] is a free operating system that can be automatically optimized and modified for just about any application or need. The developers of Gentoo believe in performance and configurability, so you can customize each element and application for your specific desktop needs and preferences. Gentoo Linux depends on the Portage package manager, which "… provides compile-time option support (through USE flags), conditional dependencies, pre-package installation summary, safe installation (through sandboxing), uninstallation of software, system profiles, and configuration file protection" [8].

TWO TERRIFIC DISTROS

DOUBLE-SIDED DVD!

Defective discs will be replaced. Please send an email to cs@linuxpromagazine.com.

ADDITIONAL RESOURCES

[1] Android-x86 open source project: *http://www.android-x86.org*

[2] Remix OS troubleshooting: *http://support.jide.com/hc/en-us/categories/202520967-Remix-OS-for-PC*

[3] Dual-boot Remix OS on Ubuntu: *https://www.youtube.com/watch?v=FywpsFMM2Fg*

[4] XDA Developers forum: *http://forum.xda-developers.com/remix/remix-os/remix-os-installation-rooting-t3293769*

[5] Installing Remix OS in VirtualBox: *https://www.youtube.com/watch?v=1t5kECk_U_Q*

[6] Dual-boot Remix OS on Windows: *http://www.jide.com/remixos-for-pc*

[7] Gentoo Linux: *https://wiki.gentoo.org/wiki/Main_Page*

[8] Gentoo FAQs: *https://wiki.gentoo.org/wiki/FAQ*

NEWS

Updates on technologies, trends, and tools

Mirantis Joins Hands with SUSE to Offer RHEL Support; Red Hat Not Happy

The OpenStack company Mirantis has joined forces with SUSE to offer support for SUSE Linux Enterprise Server on Mirantis OpenStack. Under the collaboration, SUSE and Mirantis will work together to validate, optimize, certify, and support SUSE Linux Enterprise Server for use with Mirantis OpenStack Software.

The most interesting aspect of the collaboration is that the two companies will also offer support for Red Hat Enterprise Linux (RHEL) and CentOS, in addition to SLES. Red Hat offers its own OpenStack distribution and doesn't support Mirantis OpenStack.

In an interview, SUSE told us that the support offer is based on SUSE Linux Enterprise Server with Expanded Support, which covers RHEL and CentOS. Mirantis already supports Ubuntu, and this partnership adds RHEL, CentOS, and SLES, thus offering a one-stop shop for all major Linux enterprise distros.

Red Hat has raised questions about such support and told us that any claim from Mirantis and SUSE that they can provide support for another company's offerings not only makes no sense but would be confusing and potentially dangerous for customers.

Mirantis will offer priority subscriptions in single or three-year terms and include security and other updates, plus 24x7x365 email and phone-based support with guaranteed one-hour response time. Support is available for current versions of SUSE Linux Enterprise Server, RHEL, and CentOS.

© Thomas von Stetten; Fotolia.com

Mirantis to Refactor OpenStack Fuel for Kubernetes

Mirantis is collaborating with Google and Intel to refactor the Fuel lifecycle management tool to use Kubernetes as its underlying orchestration engine. Fuel is a lifecycle management component for OpenStack.

Mirantis told me in an interview that they are using some of the work done by the CoreOS team on Stackanetes, which enables developers to deploy OpenStack on Kubernetes. "With the emergence of Docker as the standard container image format and Kubernetes as the standard for container orchestration, we are finally seeing continuity in how people approach operations of distributed applications," said Mirantis founder and CMO, Boris Renski. "Combining Kubernetes and Fuel will open OpenStack up to a new delivery model that allows faster consumption of updates, helping customers get to outcomes faster."

Mirantis is also planning to become an active contributor to the Kubernetes project to better support it. The company has also joined the Cloud Native Computing Foundation as a Silver member to collaborate with stakeholders on developing cloud native applications and services.

Linux on Windows 10 Poses a Security Risk

As the instances of Linux virtual machines are increasing on Microsoft Azure, Microsoft is looking at Linux as a development platform. To enable sysadmins and developers to manage their Linux machines from Windows, without having to resort to a VM, the company worked with Canonical to bring Ubuntu's version of the Bash shell to Windows 10. To achieve this, Microsoft has built a new subsystem within Windows called the Windows Subsystem for Linux (WSL). Ubuntu for Windows runs on top of the WSL infrastructure to offer Linux developer tools on Windows, but according to Crowdstrike chief architect Alex Ionescu, this design is creating some serious security issues.

Ionescu, who delivered a talk on WSL issues at the recent BlackHat security conference, has already reported his findings to Microsoft, and some of the issues have already been fixed. In an interview with eWeek, Ionescu said, "There are a number of ways that Windows applications could inject code, modify memory, and add new threats to a Linux application running on Windows.'

Ionescu also added that the Linux environment running in Windows is less secure because of compatibility issues with the host operating system. He noted that Microsoft's whitelisting service for Windows application, AppLocker, doesn't work with Linux applications.

© Sergey Nivens, 123RF.com

Canonical Joins Document Foundation Advisory Board

Canonical, the parent company of Ubuntu, has joined the advisory board of The Document Foundation (TDF). TDF is the organization that manages and develops the free and open source office suite LibreOffice.

"We are extremely pleased to become a member of the LibreOffice Project Advisory Board and having the opportunity to provide our guidance and insights to help improve LibreOffice for users around the world," said Will Cooke, Desktop Engineering Manager for Canonical.

The Advisory Board represents the sponsors of TDF and offers advice, guidance, and proposals to the Board of Directors.

LibreOffice has been the default office suite on Ubuntu since its early days. The upcoming release of LibreOffice 5.2 will be one of the first applications to be available through the Canonical-sponsored Snappy package management system.

Telegram Accounts Hacked in Iran

The messaging service Telegram has come under fire again. This time, accounts of dozens of Iranian Telegram users have been compromised. Hackers were also able to access more than 15 million phone numbers and accounts exploiting the API of the service.

Independent researchers Claudio Guarnieri and Collin Anderson discovered that the Telegram breach in Iran was conducted through SMS messages. Telegram uses SMS messages to send a verification code when a new device is set up. Hackers gained access to a target's SMS message and then obtained code to add their device to the target's account. Once the device was added to the account, they gained complete access to that account, including the chat history.

According to various reports, Rocket Kitten, an Iranian hacking group, is behind this breach. Researchers also hinted at the involvement of the Iranian government and cellular companies, because the service is heavily used by activists, journalists, and citizens who want to bypass the government's control. Iran has more than 20 million Telegram users.

Telegram downplayed this hack and told users to "stay calm and Telegram." In a blog post, the company confirmed the hack: "Certain people checked whether some Iranian numbers were registered on Telegram and were able to confirm this for 15 million accounts. As a result, only publicly available data was collected and the accounts themselves were not accessed. Such mass checks are no longer possible since we introduced some limitations into our API this year. However, since Telegram is based on phone contacts, any party can potentially check whether a phone number is registered in the system. This is also true for any other contact-based messaging app (WhatsApp, Messenger, etc.)."

The company also reminded users to use two-factor authentication instead of SMS verification. SMS verification is not trustworthy, and if you rely on end-to-end encryption of your messaging, this breach is a reminder to be aware of other factors, such as SMS authentication, that can compromise your account.

This is not the first time Telegram has been under fire. Last year, security researchers pointed out many security issues affecting the service.

Jeep Cherokee Hacked Again

Two cybersecurity researchers Charlie Miller and Chris Valasek, who hacked Jeep Cherokee last year, have demonstrated yet another hack of the vehicle, taking almost complete control of the car. Unlike their previous hack, which was executed remotely, the new hack requires physical access to the car.

Miller and Valasek plugged in their laptops to the OBD (the diagnostic port of the car) and gained control of almost every feature that can be controlled via computer. Once they were connected to the car, they updated the firmware for ECU (electronic control unit) and enabled certain features that were disabled for security reasons. This allowed them to take control of the steering wheel, even when the car was not in reverse mode or if it was driven at any speed. They were also able to apply the parking brake and change cruise control settings.

In a statement to Wired, Fiat Chrysler (FCA) said, "While we admire their creativity, it appears that the researchers have not identified any new remote way to compromise a 2014 Jeep Cherokee or other FCA US vehicles."

Although this hack did require physical access to the car, it does demonstrate a potential for hacking cars through the OBD dongles that are offered by insurance companies. It also demonstrates that eventually someone may find a remote hack to gain access to these controls.

© Vasyl Nesterov, 123RF.com

FCA further added, "It is highly unlikely that this exploit could be possible … if the vehicle software were still at the latest level." Miller and Valasek disputed the company's claim and stated this hack has nothing to do with the infotainment system that was patched by the company.

Miller and Valasek shared the findings of their research at the Black Hat security conference on August 3-4.

Miller and Valasek entered the limelight a year ago when they remotely hacked a Jeep Cherokee. The duo was able to disable the transmission and brakes of the car while it was driven by a Wired reporter on the road. They were also able to take control of the steering wheel while the car was in reverse or parallel parking mode. Post hack, the duo was hired to work at Uber's Advanced Technology Center in Pittsburgh.

Pokemon Go Hacked by OurMine

The phenomenal success of Pokemon Go is also attracting cyber criminals. Pokemon Go experienced severe server issues on July 16; however, the company didn't mention an attack.

According to PCMag, the OurMine hacker group was behind the server issue. Members of OurMine said via email that they were just trying to protect the company's servers. "We wrote we will stop the attack if any [Niantic] staff talked with us, because we will teach them how to protect their servers," a member of OurMine told PCMag.

There is another group called PoodleCorp that is claiming a DDoS attack on Pokemon Go servers. PoodleCorp is reportedly planning another attack on August 1.

The popularity of the game has raised concerns around privacy and security. Senator Al Franken (D) of Minnesota penned a letter to the game developers and expressed his concerns. He wrote, "I am concerned about the extent to which Niantic may be unnecessarily collecting, using, and sharing a wide range of users' personal information without their appropriate consent."

© Lee Chee Keong, 123RF.com

Popular blogger and Entrepreneur in Residence at Upload VR, Robert Scoble, also expressed his concerns on Facebook, "I haven't seen a single person worrying about the privacy implications of Pokemon Go. Yes, we have gone over the freaky line that Shel Israel and I laid out in our book *Age of Context*. Five years ago, this game would have caused a huge privacy panic. This thing tracks our children. Think about that!"
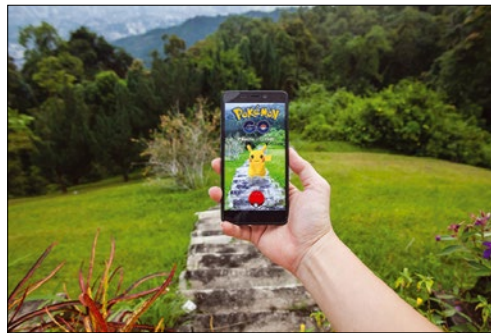
Ubuntu Forums Breached Again

Canonical reported on July 14 that Ubuntu forums were breached. Attackers were able to steal usernames, email addresses, and IP addresses of more than two million users.

This is the second time Ubuntu forums have been breached. The last breach happened in 2013 when attackers stole email addresses, passwords, and usernames of members. Despite its claims of being an open source Linux company, Canonical is running its forums on proprietary vBulletin software. But Canonical is not alone, Fedora and SUSE also use vBulletin for their forums.

Ubuntu forums were hacked because admins at Canonical didn't update the forum software. Security expert Graham Cluley wrote on his blog, "What a goof. If you don't patch the software running on your website, don't be surprised if a hacker compromises your system and makes off with your customer's data."

This is the second high-profile breach in the desktop Linux world. The Linux Mint site was hacked earlier this year.

Changing from a Samba classic domain to Samba 4

Upgrade or Wait?

Samba 4 has been around for more than three years, but some users still shy from it. If you are still sitting on the fence, this tour through some of the new features and capabilities might help you decide whether it is finally time to upgrade.

By Fabian Melters
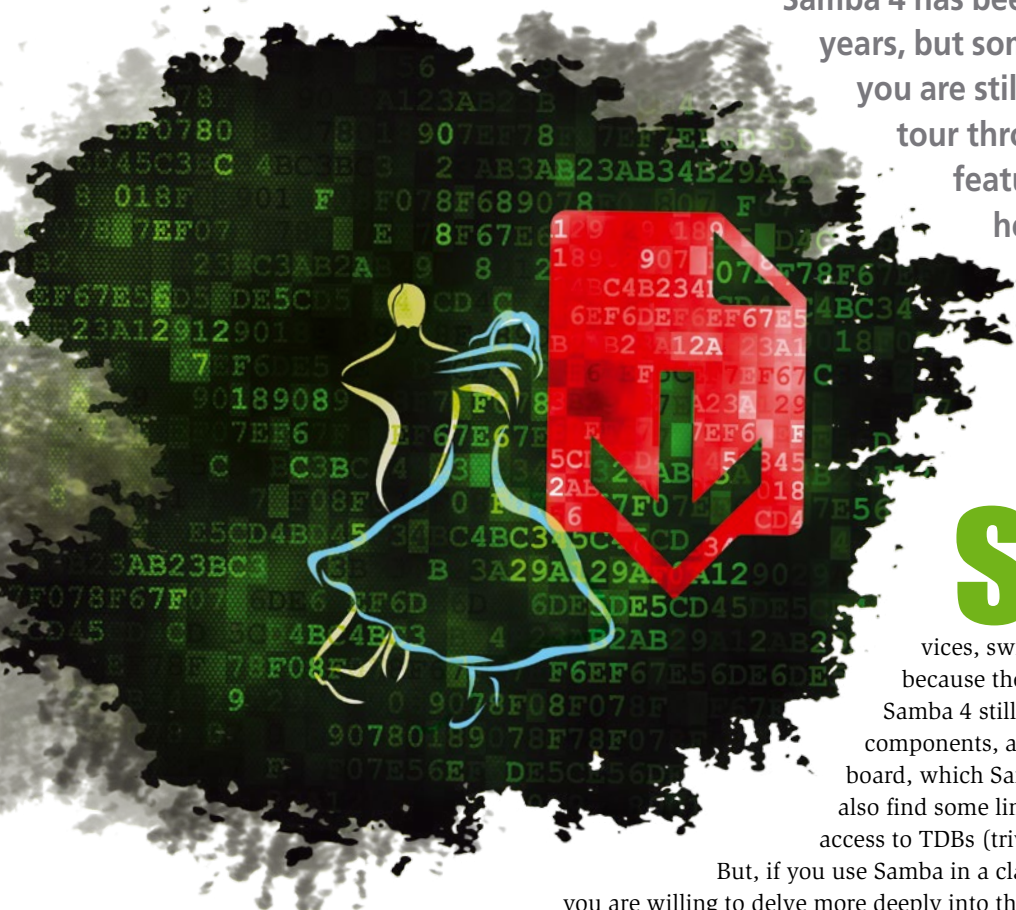
Samba [1] is the tool of choice for providing Windows-like file and print sharing services on a Linux server. If you only need basic file and print services, switching to Samba 4 it not worth the effort because the new Samba is not so much different. Samba 4 still contains the `smbd`, `nmbd`, and `winbindd` components, although it also has the `samba` service onboard, which Samba needs for the new features. You will also find some limitations. For instance, Samba 4 reduces access to TDBs (trivial databases) to a minimum.

But, if you use Samba in a classic Windows NT4 domain setting, and if you are willing to delve more deeply into the Windows world, some of the new Samba 4 features might come in handy.

What's New?

The main feature of Samba 4 is the possibility of using a Samba server as a full-fledged replacement for a Windows AD domain controller. Samba 4 supports Windows environments as of Windows 2000.

An LDAP server developed for Samba and integrated into the solution itself assumes the Active Directory role. Samba 4 relies on the built-in Kerberos KDC (Kerberos Key Distribution Center) to support Kerberos authentication via its usual ticket system.

DNS still plays a central role in a Windows AD domain, and you can set up Samba 4 with two possible DNS server roles. The Samba 4 binary includes a DNS server that is part of Samba 4. If necessary, admins can call on the DLZ (dynamically loadable zones) module to implement a Bind server. Both solutions support the typical maintenance of DNS records that is familiar to Windows administrators. The official recommendation is to use the built-in DNS and only change to Bind if necessary.

The correct system time is also important for Windows domains. For example, Kerberos relies on correct timestamps to avoid replay attacks. Samba 4 keeps time by accessing the well-known NTP daemon. (Btw: Windows 2000 clients do not behave as an NTP server would expect, so Samba 4 cannot act as an AD domain controller for Windows 2000 systems.)

COVER STORIES

Management

Windows admins can customize the entire configuration using the Microsoft Management Console (MMC). MMC is the standard tool for managing a Windows AD domain controller (see Figure 1). On the Linux side, admins manage current Samba versions with the help of the new `samba-tool` [2]. With `samba-tool`, you can create or delete users and groups or trigger a `classicupgrade` from version 3 to 4. Table 1 shows the options.

Samba 4 has a programming interface for Python. Admins and system integrators use this interface to seamlessly customize the software for their environments. Many Samba 4 tools (including `samba-tool`) use Python and rely on this interface.

Versions

Major and minor versions of Samba 4 appear regularly. Major changes end up in major versions with numbers such as 4.0, 4.1, 4.2, and so on. Minor changes are incorporated in the minor versions, with version numbers like 4.1.1, 4.1.2, 4.2.1, or 4.2.2. The latest stable release is version 4.4.3 [3].

The major releases 4.1, 4.2, 4.3, and 4.4 improve performance and substantially expand the feature list. Since version 4.1, for example, the client tools now also work with the SMB 2 and 3 protocols. In addition, Samba 4.1 enables server-side copy actions.

Release 4.2 saw the CTDB (Cluster Trivial Database) enter the Samba tree. The CTDB lets you run Samba file servers in the form of clusters. A new tool called Samba Registry Editor lets you crawl the Samba registry.

The 4.3 and 4.4 releases improve existing features and add enhancements to some of the tools. Version 4.4 impresses with better performance, especially with asynchronous flush requests. When clients ask the server to write unsecured content to disk, this write operation is done in an asynchronous manner. The operation therefore blocks any other processes. Support for SMB-3-multichannel is still considered experimental, however. This feature allows the client to build multiple transport connections in an authenticated SMB session, which improves both fault tolerance and data throughput because the file can be transferred in parallel over multiple network connections.

Last but not least, some of the changes relate to the configuration in the /etc/
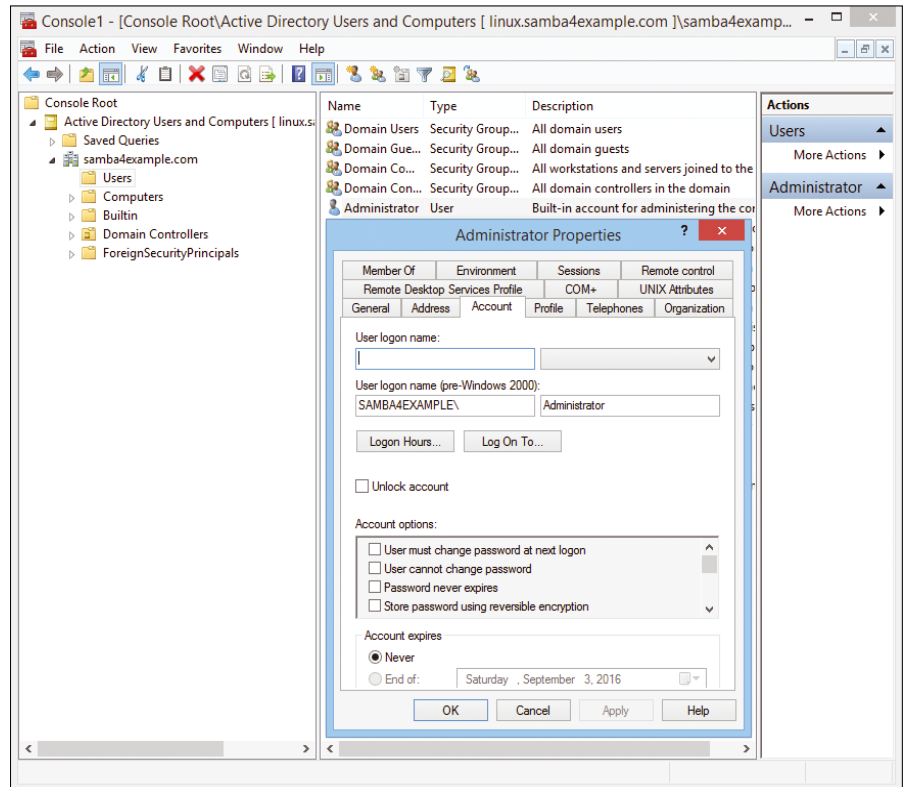
Figure 1: The Windows admin still uses the MMC process user data even if Samba 4 is in use.

samba/smb.conf file. The developers have removed 14 parameters, changed the default setting for seven, and added 37 new parameters [4].For those who like more detail, have a look at the release notes for the major releases 4.1 [5], 4.2 [6], 4.3 [7], and 4.4 [8].

Bottlenecks

The two biggest performance issues when comparing Samba 4 and a Samba classic domain are network drives and the do-

TABLE 1: samba-tool Commands

Command	Function
dbcheck	Check the local AD database for errors
delegation	Manage delegations
dns	Manage the domain name service (DNS)
domain	Manage domains
drs	Manage the directory replication service (DRS)
dsacl	Manage access control lists (ACL) for domain services
fsmo	Edit roles for flexible single master operations (FSMO)
gpo	Manage Group Policy objects
group	Manage groups
ldapcmp	Compare two LDAP databases
ntacl	Manage NT ACLs
processes	List processes (for debugging on systems without setproctitle)
rodc	Manage the read-only domain controller (RODC)
sites	Site management
spn	Manipulate identifiers of service instances (service principal names)
testparm	Check config file for syntax errors
time	Retrieve the timestamp on the server
user	User management
vampire	Synchronize a remote AD domain with the local server

main controller. Samba 4 can boost performance of network drives because it supports SMB 3, which Microsoft launched with Windows 8 and Windows Server 2012.

As a domain controller, Samba 4 tends to be worse in terms of data throughput. Practical experience shows that the built-in LDAP server does not compare well with OpenLDAP. But a slower LDAP server only really causes problems when different services use the LDAP connection – and not only to authenticate. This issue can cause bottlenecks if admins also operate DHCP servers, groupware, and mailing lists derived from LDAP.

Samba 4 is also much more memory-hungry than its predecessor. This is especially noticeable when many users are loading the network drives. Admins need to provide enough RAM to maintain a consistent level of performance.

NetBIOS and Browsing

Samba 4 still does not support NetBIOS browsing on the Samba 4 domain controller. Windows clients do not see the domain controller in their network environment. In other words, if Samba 4 is defined as the NetBIOS master, no network devices appear in the network neighborhood.

Many versions of Microsoft Exchange require an LDAP feature that is still lacking in the Samba 4 LDAP, so at this time, it is only possible to add a Microsoft Exchange 2010 system on a Microsoft Server 2008 R2 to the Samba 4 domain.

As is often the case, the Samba 4 environment has several minor stumbling blocks that only stop the show in specific cases. See the Samba bug tracker [9] for details.

Choices

The first important thing is to choose a suitable version of Samba. Although it has been possible to use the Active Directory domain controller in Samba 4 since version 4.0, only version 4.3 promises comparatively little trouble. If you are shopping for a distribution, you should consider Debian Jessie, which incorporates a reasonably current version of Samba, version 4.2.10. But if you also need star and hub replication for multi-site scenarios via KCC, you will need version 4.3. Older versions only allow mesh replication.

Using the Debian Sid development edition will put you closer to the Samba pulse, but a development version is out of the question for many companies. Most other distributions in the enterprise come with rather obsolete packages, including many bugs that more recent versions of Samba 4 fix. If the distribution you use does not come with a current Samba 4 version, admins still have the opportunity to buy packages from Sernet or build from the sources.

Migration

It is a good idea to test as many steps as possible in advance using sandbox machines. Firewalls prevent the NT4 domain controller and the Active Directory domain controller from getting in each other's way during the migration. But if they do so, some manual attention is required, because the admin needs to convert each individual computer.

One fairly typical scenario is that the operator of NT4 DC runs the domain controller together with the file server on the same machine. Samba 4 won't let you do the server to this dual role, so the best approach is to let the file server run on the ex-

isting server and set up the domain controller on a virtual machine or another computer.

You also need to keep in mind that groups and users may use the same name in AD environments. Last but not least, administrators need to change the LDAP server integrated into Samba 4 via `/etc/samba/smb.conf` from `localhost` to the IP address of the old LDAP server. Use `samba-tool domain classicupgrade` [10] to switch to a full-fledged Samba 4 AD domain controller. Backing up configuration files is recommended, but the upgrade works with all password back ends, that is, `smbpasswd`, `tdbsam`, and `ldapsam`.

Admins should also understand that the change from a NT4 domain controller to Samba 4 is a one-way street; you cannot easily roll back. A methodical and granular approach is thus useful. Beginners might find it easier to just start with Samba 4 in the first place.

Conclusions

All told, Samba Version 4 is a viable alternative to Windows AD domain controllers, and it provides most of the features you would get with a Windows system. The performance should be absolutely adequate in environments without special configurations.

If you do not rely on the functionality of an AD domain controller and have deployed a Samba Classic domain without trouble for years, the only benefit you might actually experience after the migration is better network drive performance with Samba 4. ∎∎∎

▮ INFO

[1] Samba: *https://www.samba.org/*

[2] User and group management: *https://wiki.samba.org/index.php/User_and_Group_management*

[3] Samba 4.4.3 Release Notes: *https://www.samba.org/samba/history/samba-4.4.3.html*

[4] Samba 4.0.0 Release Notes: *https://www.samba.org/samba/history/samba-4.0.0.html*

[5] Samba 4.1 Release Notes: *https://www.samba.org/samba/history/samba-4.1.0.html*

[6] Samba 4.2 Release Notes: *https://www.samba.org/samba/history/samba-4.2.0.html*

[7] Samba 4.3 Release Notes: *https://www.samba.org/samba/history/samba-4.3.0.html*

[8] Samba 4.4 Release Notes: *https://www.samba.org/samba/history/samba-4.4.0.html*

[9] Open bugs in Samba 4.1: *https://bugzilla.samba.org/buglist.cgi?bug_status=NEW&bug_status=ASSIGNED&bug_status=REOPENED&order=bug_id&product=Samba%204.1%20and%20newer&query_based_on=&query_format=advanced*

[10] Migration of Samba 3 NT domains to Samba 4 AD domains: *https://wiki.samba.org/index.php/Migrating_a_Samba_NT4_domain_to_a_Samba_AD_domain_%28classic_upgrade%29*

▮ AUTHOR

Fabian Melters works for Linux Information Systems as a project manager, consultant, developer, and IT/security system engineer with a focus on IT security, firewall, monitoring, and mail gateways.

This is an advertisement poster.

Accelerate Your Android Development!

From mobile app development training to embedded Android and the Internet of Things, AnDevCon offers the most comprehensive program with countless sessions and networking opportunities. Roll-up your sleeves, dive into code, and implement what you learn immediately.

ÄnDevCon
The Android Developer Conference
Nov. 29 - Dec. 1, 2016
San Francisco Bay Area
Hyatt Regency Burlingame

Take your Android development skills to the next level!

- Choose from more than 75 classes and in-depth tutorials
- Meet Google Development Experts
- Network with speakers and other Android developers
- Check out more than 50 third-party vendors
- Women in Android Luncheon
- Panels and keynotes
- Receptions, ice cream, prizes and more!

"Simply the best Android developer conference out there! A must-go if you do Android development."
—Florian Krauthan, Software Developer, Hyperwallet

www.AnDevCon.com

AnDevCon™ is a trademark of BZ Media LLC. Android™ is a trademark of Google Inc. Google's Android Robot is used under terms of the Creative Commons 3.0 Attribution License.

A BZ Media Event

Run Samba in clustered mode with Ceph

Double Sure

Fail-safe is a massive topic for file server admins. Thanks to the CTDB and Ceph, you can put Samba in a cluster with minimal complications. *By Martin Loschwitz*

The popularity of Samba means file server admins have to think about how they can protect the service against loss. Samba is now mature and runs without any problems in most cases, but if the server on which Samba is running crashes, the service is no longer available.

The Samba developers are aware of the need for some fault tolerance and have responded to the problem with a genuine cluster option. Samba's cluster mode means you can use several Samba servers to process incoming requests. A single Samba server crash will not stop the show because other servers in the cluster will keep working.

Configuring Samba's cluster mode is not entirely intuitive, especially considering that the Samba cluster implementation has changed radically several times in the past few years. This article offers a quick look at high availability with Samba.

The Challenge

Why is a Samba cluster such a challenge? A little excursion into the world of storage theory will offer some answers. In particular, the issue of locking is very important. How does the application handle concurrent access to the same file? "Application," in this case, can mean a simple filesystem on a disk or a complex application. In any case, just imagine the chaos if two clients simultaneously access the same file and change parts of it. The file would end up corrupted, and neither client A nor client B could do anything with the contents.

Various filesystems have tried practically every conceivable solution for file locking: Older filesystems rigorously deny access to a file if it is already open. Modern filesystems follow the principle that the last write wins and determines the contents of the file.

Because Samba offers a network filesystem, it also has internal locking functions. Samba uses the TDB (Trivial Database) database format for storing internal metadata. One of the most important databases is `locking.tdb`, which tracks which client is currently accessing which file.

Samba relies on opportunistic locking, which means a client tells the server that it has claimed exclusive access rights to a file on the Samba share for itself. Once the Samba server has complied with the request, it writes a corresponding note to `locking.tdb` and stops other clients from accessing the same file.

As long as the process is limited to a single instance of Samba, everything works fine: The single Samba server can reliably assume that its version of `locking.tdb` is authoritative.

But a clustered configuration adds a challenge: Multiple Samba instances need to sync the contents of their `locking.tdb` files with each other. The cluster must therefore have some means for managing client access to files on the Samba volume.

years at the beta stage, but the last version of Ceph "Jewel" promises a higher level of maturity: CephFS is suitable for the production operation, according to the developers.

Three servers are available in the following example of Ceph: Alice, Bob, and Charlie – each of these servers has a hard drive that it contributes for the Ceph object store. Although the performance benefits of Ceph are best realized when the cluster runs on real hardware, you can easily emulate this configuration on virtual machines if you only want to try things out.

Even the most attractive Samba cluster will be no help if you ignore fundamental rules of high availability (HA). Basically, an HA cluster with Samba faces the same challenges that all other services on a server need to take on: Clustering at the software level only checks one box on the list. The loss of infrastructure that is not controlled by Samba can still trip Samba up.

Network and power are the two classic infrastructure issues you'll need to address: Several Samba servers in the combined cluster are good, but if they are all connected to the same electrical circuit and the circuit fails, both servers are dead. The problem is the same for Ethernet: If all nodes in the cluster are connected to the same switch and it fails, the Samba service is still available, but its clients can no longer reach it.

Creating the Necessary Infrastructure

The degree of redundancy depends on the budget for the project. Redundancy at the power and network levels can cause significant additional costs, because you'll need to duplicate many components. Admins face a compromise: The more parts you make redundant, the lower the risk of failure, but the setup is more expensive.

Step 1: Get Ceph Ready for Action

The first step toward setting up Samba on a cluster is to determine the data storage solution; in this example, the solution is CephFS on a Ceph cluster (for alternatives, see the box entitled "Plan B for Ceph and Samba"). It is very good idea to use the

The solution for this problem, say the Samba developers, is CTDB (Clustered Trivial Database), an extension of TDB that lets many instances of Samba dynamically share TDB content.

Requirements for Clustered Samba

A few years ago, the option for a cluster file server was some form of clustered filesystem: solutions such as GFS or OCFS2 (Oracle Cluster Filesystem 2) could manage cluster-wide access to the same filesystem in a NAS share connected via iSCSI. But solutions of this sort required a cluster manager, preferably Pacemaker, and configuring and managing Pacemaker can be a very complicated task – especially when you are using it with GFS or OCFS2.

Luckily, distributed storage solutions have led to a simpler approach. Distributed storage tools such as GlusterFS and Ceph work differently: A large filesystem comprises many small segments on the participating servers, and consistency issues are addressed internally. Access occurs through designated, independent mechanisms via simple interfaces. In truth, distributed storage is no less complex than Pacemaker with OCFS2, but it does a better job of hiding the complexity. The barrier to entry is thus lower.

Two rival distributed storage solutions dominate the market, and both are sponsored by Red Hat: On one hand, GlusterFS offers a classic distributed filesystem; on the other, Ceph is an object store that can offer its contents in the form of a POSIX-compatible filesystem, CephFS. CephFS was stuck for several

PLAN B FOR CEPH AND SAMBA

If you are looking for an alternative to the configuration described in this article, you can combine Ceph and Samba in other ways. For instance, Red Hat's developers have written a VFS module for Samba that is coupled directly to the RADOS library and can talk directly to the object store.

The advantages of this VFS solution are obvious: On one hand, the need for a local CephFS mount on the Samba server is removed. As a result, the deadlock problem mentioned previously in the context of releasing storage elegantly disappears. Additionally, it is no longer necessary to run metadata servers because you are not using CephFS.

The VFS module for Samba is included in all versions of Samba from 4.2 (Figure 1). If you find a more recent package in your distribution, you can the opt for the vfs_ceph module. Look online for details of the drivers and the necessary configuration [2].

Figure 1: An alternative to detouring via CephFS; the VFS module for Ceph is included in the current versions of Samba (from 4.2).

current version of Ceph, which was Jewel 10.2.0 when I wrote this article, because the developers only guarantee stability as of this version of CephFS. In this example, I used Ubuntu 16.04; the Ceph project already has pre-built packages with the required Ceph version [1].

You can use `ceph-deploy` to set up the Ceph configuration. This tool prepares the three computers for Ceph, installs Ceph, and also lets the admin activate the disks in Ceph. The first step, then, is to install `ceph-deploy`. Although a package with the tool is included in Ubuntu, I recommend using the `ceph-deploy` version from the Red Hat repository (Figure 2). You need to enable the repository up front by typing

```
wget -q -O- 'https://download.ceph.com/keys/release.asc' | ⌐
    sudo apt-key add -
```

for the GPG key and then

```
sudo apt-add-repository ⌐
    'deb http://download.ceph.com/debian-jewel/xenial main'
```

for the packages themselves. Then, install the tool on your system with:

```
apt-get update && apt-get install ceph-deploy
```

You only need to run these steps on one of the three systems.

Preparing the Servers for Ceph
For Ceph and `ceph-deploy` to work, you need a working SSH connection. On each host, you should also have a user named *ceph* who runs `ceph-deploy`. The user should also be able to use `sudo` without a password to run the commands on the systems as root. The following commands will do the trick:

```
echo "ceph ALL = (root) NOPASSWD:ALL" | ⌐
    sudo tee /etc/sudoers.d/ceph
sudo chmod 0440 /etc/sudoers.d/ceph
```

The `ceph-deploy` command uses SSH to execute commands within the cluster. To avoid the need for many passwords, the *ceph* user should be able to connect freely between the servers using SSH without a password.

The `ssh-copy-id` command copies the public part of an SSH key to a remote host. The default username should also be set to *ceph* in the `~/.ssh/config` file on the other servers in the Ceph cluster, which will help you avoid `ceph-deploy` trying to log in as root.

The three computers and their IP addresses on all three hosts must be available in `/etc/hosts`. It makes sense to make one of the three nodes the master and use it for the `ceph-deploy` calls, although this has no effect on the functionality of Ceph.

Rolling Out the Ceph Cluster
Once `ceph-deploy` is ready to run, the other steps of the Ceph installation are almost automatic. The `install` command in `ceph-deploy` makes sure the necessary packages for Ceph exist on all specified hosts. The command

```
ceph-deploy install --release jewel alice bob charlie
```

run on the imaginary master host automatically installs Ceph on all three computers.

The cluster setup follows. Just as a reminder: Ceph consists of three components: Object Storage Devices (OSD), Monitoring Servers (MON), and Metadata Servers (MDS). The OSDs are the hard drives that the cluster requires. The MDSs extend the Ceph object store to include POSIX capabilities for use as a filesystem. The MONs are the most important components of Ceph: They monitor the quorum and ensure cluster operation.

When rolling out Ceph, the MONs are the first components you need to turn to. For installation in a new cluster, `ceph-deploy` has the `new` command. On the master server,

```
ceph-deploy new alice bob charlie
```

automatically rolls out a new Ceph cluster on the three hosts; At the end of the process, `ceph-mon` is running on each of the hosts. Once `ceph-deploy` has completed its work, the `ceph -w` command on the first of the three hosts will identify a working cluster.

The next step is to set up the disks: On Alice, Bob, and Charlie, `/dev/sdb` is intended for Ceph. The command

```
ceph-deploy create alice:sdb
```


Figure 2: The current version of `ceph-deploy` is significantly newer than the one delivered by Ubuntu.

(similar for Bob and Charlie) adds the disks to the cluster. If you want to outsource the OSD's internal journal to a fast solid-state drive (SSD), you can define the target partition by tagging :<partition> to the end of the command, as in

```
ceph-deploy create alice:sdb:/dev/sdf1
```

where /dev/sdf1 is a partition on a SSD.

Because CephFS will be used in the cluster, you'll need metadata servers. You install the metadata servers with the ceph-deploy command, as well:

```
ceph-deploy mds create alice bob charlie
```

Another call to ceph -w shows a new situation: Now you have a Ceph cluster with three MONs, three OSDs, and three MDSs, and it is ready for operation (Figure 3).

Incidentally, if you want to operate this cluster later on, you will need physical disks. The usual Ceph tips for redundancy and performance apply: You should have three individual servers, and if you want to out-source the OSD journals to fast SSDs, you will want to put more than four OSD journals on a single SSD. The throughput is a practical measure of a Ceph cluster's quality – a fast connection of each Ceph node certainly won't hurt.

Step 2: Preparing for Samba

The next step is to configure Samba so that it uses CTDB and accesses CephFS. (Operating Samba on the Ceph cluster nodes is a tempting proposal, but the

Ceph developers strongly recommend you avoid the potential loopback problems that could result from enabling a CephFS filesystem mount on a host that is part of the Ceph cluster itself.)

Samba will run on separate hosts and access CephFS remotely. The other servers in this configuration answer to the names of Daisy and Eric.

You first need a CephFS mount on the Samba systems. Ceph relies on the built-in authentication mechanism, CephX, which ceph-deploy also enables. For the mount to work, you need the password of an active CephX user. In this article, I assume that

Figure 3: The list of available resources after setting up of the Ceph cluster: three MONs, three OSDs, and three MDSs, all waiting for requests.
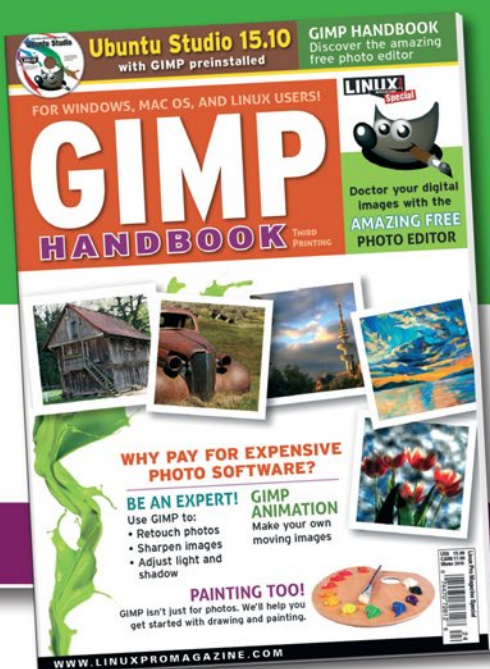
access relies on the rights of the admin user *admin*. The Ceph documentation explains the essentials of user management [3].

The password of the *admin* user is found on the master server in `/etc/ceph/ceph.client.admin.keyring`; it is the entry that follows behind `key =:` in this example, `AQCj2YpRiAe6CxAA7/ETt7Hc19IyxyYciVs47w==`. This key belongs in a separate file with a freely selectable name, such as `/etc/ceph/admin.secret`. Now you can mount CephFS using `/mnt/samba`:

```
sudo mount -t ceph IP_address:6789:/ ⏎
    /mnt/samba -o name=admin,secretfile=/etc/ceph/admin.secret
```

The IP address should be the IP address of a MON server, such as the local IP address of Alice. You can also add the mount entry to your `/etc/fstab` file:

```
IPaddress:6789:/ /mnt/samba ceph ⏎
    name=admin,secretfile=/etc/ceph/admin.secret,noatime 0 2
```

After you reboot the system, CephFS is immediately available under `/mnt/samba`. The entry and the keyfile should be present on all hosts that want to mount a CephFS filesystem.

Step 3: Using CTDB

To make CTDB available, you must enable cluster mode explicitly when compiling Samba. All current distributions come with cluster-capable Samba in a sufficiently recent version – CTDB requires version 4.2 or newer of Samba.

At least four parameters must exist in your `smb.conf` for CTDB to work:
- `netbios name=<entry>`
- `clustering=yes`
- `idmap config * : backend=autorid`
- `idmap config * : range = 1000000–1999999`

You also need to install the separate `ctdb` package, which contains all the programs related to CTDB.

Figure 4: Use the `ctdb status` **command to display information about the state of the CTDB cluster.**

Figure 5: `ctdb ping` **executes a** `ping` **command to all nodes in the CTDB cluster and displays the results.**

In addition, you need several CTDB-specific configuration files that you have to adapt to local conditions. Some required values are:
- `CTDB_NODES`, which points to a file that lists all participating nodes of the Samba cluster. The default is `/etc/ctdb/nodes`; the program expects the IP address of one of the cluster nodes in a line of the file.
- `CTDB_RECOVERY_LOCK`, which points to a file that CTDB expects in the shared storage; in this example, `/mnt/samba/lock`.
- `CTDB_PUBLIC_ADRESSES`, which is a bit complicated: CTDB expects a file containing a list of all network interfaces of each node together with the associated IPs. The syntax of the file is `IP/netmask <network_interface>`. For the example with Daisy and Eric, the file might look like:

```
10.42.0.1/24 eth0 10.42.0.2/24 eth0
```

`CTDB_PUBLIC_ADRESSES` clarifies the fact that CTDB is a lightweight cluster manager: CTDB needs the details of the IP addresses to be able to activate its IP address on a different Samba node after the failure of one node.

If the host to which an IP address from `CTDB_PUBLIC_ADRESSES` is assigned fails at any time, CTDB automatically ensures that the IP is enabled elsewhere and thus also ensures that the CIFS clients continue to receive responses to requests. The IP addresses from `CTDB_PUBLIC_ADRESSES` also need to be entered in DNS so that name resolution works.

After these steps, Samba is ready to go: In addition to the well-known services `smbd`, `nmbd`, and `winbind`, the `ctdb` service should be running also. The next step is to run the command that shows whether the CTDB setup worked:

```
ctdb status
```

Multiple nodes should show up, and the cluster should have a status of *NORMAL* (Figure 4). Then, each of the CTDB nodes can act as a single Samba server.

In the background, Samba stores data to the cluster. A built-in health check,

```
ctdb ping
```

pings all the other CTDB nodes from the current node and displays the response times (Figure 5). ∎∎∎

INFO

[1] Ceph Jewel for Ubuntu 16.04: *http://download.ceph.com/debian-jewel/dists/xenial/main/binary-amd64/*

[2] vfs_ceph for Samba: *http://manpages.ubuntu.com/manpages/xenial/man8/vfs_ceph.8.html*

[3] CephX management: *http://docs.ceph.com/docs/hammer/rados/operations/user-management/*

AUTHOR

Martin Gerhard Loschwitz works as a cloud architect at SysEleven in Berlin. He is also an official member of the Debian project and has been a Debian developer for more than 12 years.

The state of the classic NFS filesystem

Whither NFS?

The NFS network filesystem has served Unix and Linux networks for many years, but the demise of NFS inventor Sun Microsystems as an independent company has thrust NFS into a creative crisis. Will this veteran from the early days of Unix find the strength to rise again? *By Martin Steigerwald, Tim Schürmann, and Jan Kleinert*

Ever since Oracle acquired Sun Microsystems, the development of the once-omnipresent Unix network filesystem NFS has slowed considerably. Competitors such as Samba, and a new class of distributed network storage solutions, are competing with NFS for mindshare and market share within the open source community. Has NFS gone away? Not really, but it could surely use a burst of energy to regain some of the momentum it has lost to competitors.

NFS development is now the responsibility of the Internet Engineering Task Force (IETF). The current NFS version is number 4.1, which is described in RFC 5661 [1]. (RFC 5661 dates back to 2010, which gives an indication of the current level of development activity.)

The current Linux implementations [2] consist of several parts. The NFS server, the NFS filesystem, and the Sun remote procedure call (RPC) are part of the Linux kernel.
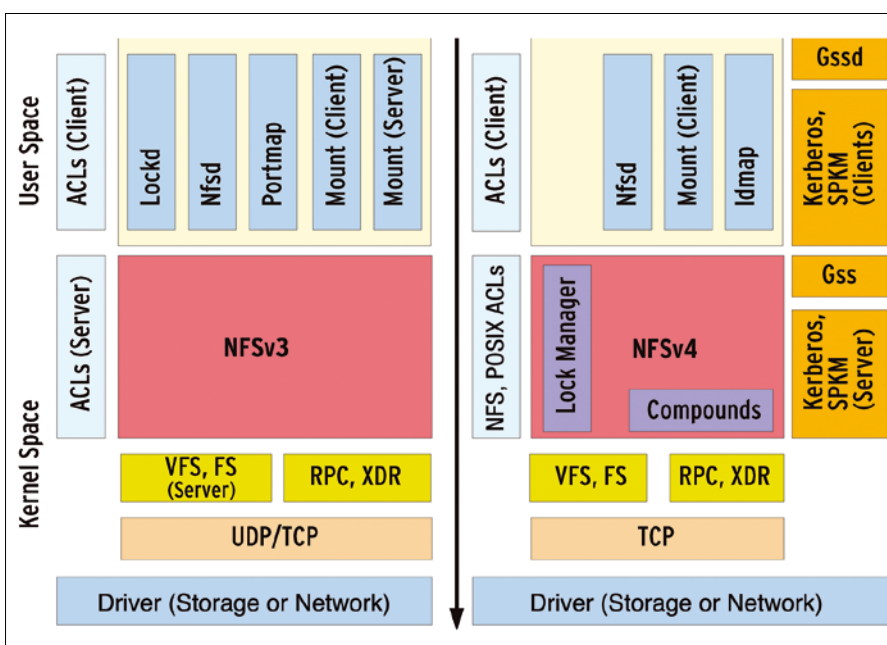
Figure 1: Structural comparison between the obsolete NFSv3 and the current NFSv4.

Today, admins will only want to deal with NFSv4. The NFSv3 architecture from the Unix heyday is not fit for today's security landscape (Figure 1); for instance, NFSv3 handles authentication client-side, blindly trusting in its security.

To help you start the current, kernel-based NFS server, some distributions offer tool packages – on Ubuntu, for instance, you'll find the `nfs-kernel-server` package. Among other things, you'll find the `exportfs` command for exporting NFS shares and matching unit files for systemd. As an alternative to the NFS server that is built into the kernel, some distributions provide the `rpc.nfsd` daemon, which runs entirely in user space. `rpc.nfsd` is no longer used much in practice.

In any case, a separate package bundles some useful command-line tools. These NFS-utils (currently version 1.3.3) are found on Ubuntu, for example, in the `nfs-common` package. It is generally worth considering how well the NFS maintainer for your choice of Linux distribution keeps pace with the utils; Debian, for example, currently uses version 1.2.8. The tools include the commands for mounting NFS shares, as well as some analysis utilities, including `nfsiostat`, `mountstat`, and `showmount`. You will even find patches for the NFS-utils [3] that retrofit support for systemd.

These patches have already reached the Git repository, but some distributions still do not include them. For example, Debian Unstable only had version 1.2.8 of the NFS-utils when this issue went to press. However, the patches are only included as of version 1.3.

Big Deficit: Poor Documentation

If you like to stay current with new features in the NFS (kernel) development, you will certainly feel the further impact of the disappearance of Sun Microsystems: The quality and quantity of documentation – or rather the lack of it – is striking. If you look around, you will stumble over some out-of-date Internet sites like the Linux NFS FAQ [2]. The Linux NFS wiki [4] is also a mix of outdated and current information.

Administrators are most likely to find good documentation offered by providers of commercial NFS-related services, such as NetApp [5]. Panasas is mainly involved with the parallel-storage version of NFS known as parallel NFS (pNFS). The company provides its own site [6] with information and even training videos on pNFS technology.

NFS developers and users discuss events such as BakeAThon and Connectathon on the mailing list for Linux NFS [7]. The Nfsv4bat.org website offers presentation slides and even some videos of the two events.

Development Status of the Software

Both the NFS wiki and many other NFS-related sites lack information as to which NFS functionality is available with which kernel version. It is thus a Sisyphean task to reveal the current state of development of the NFS server in the kernel, the client programs, and your own choice of Linux distribution.
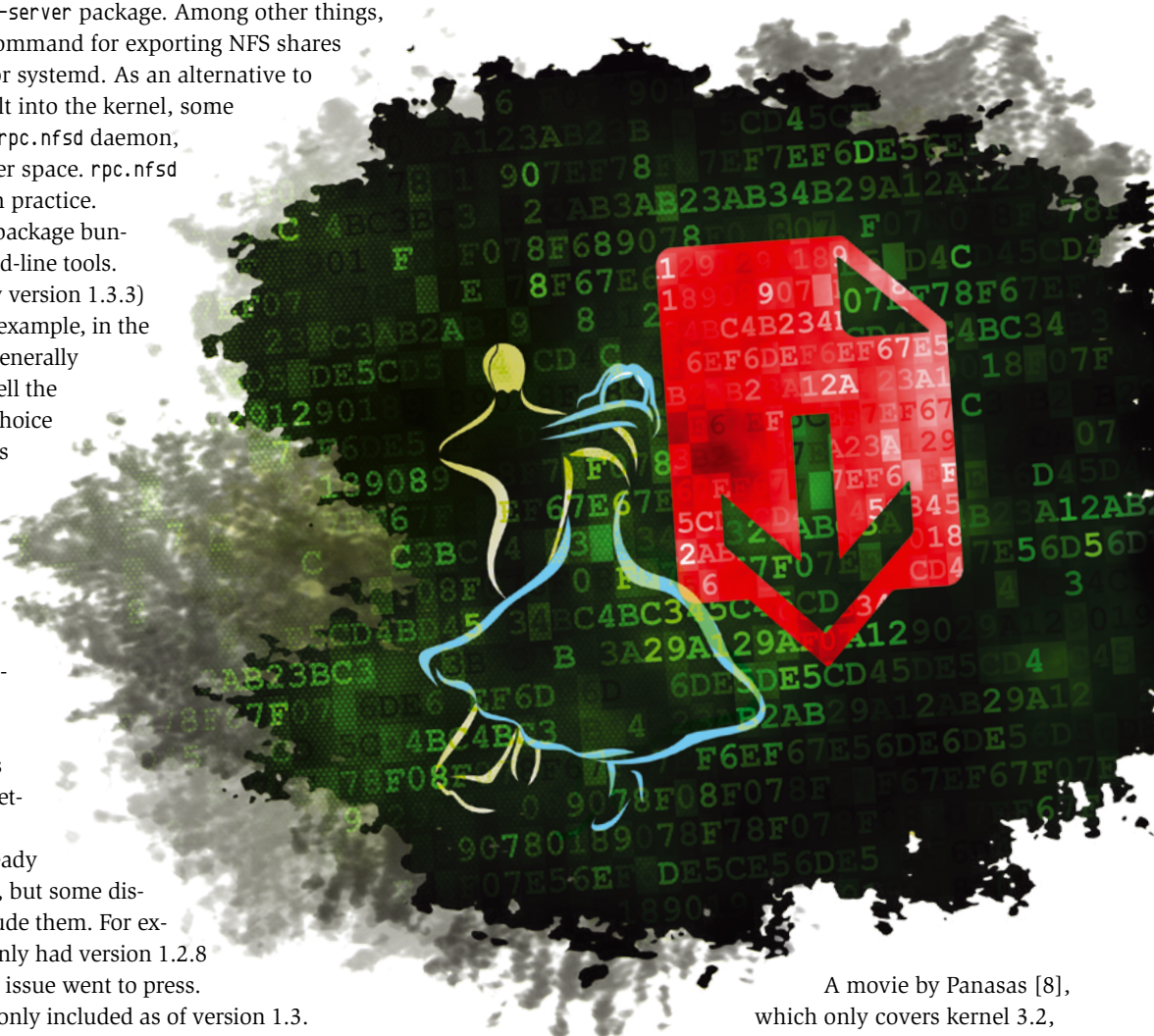
A movie by Panasas [8], which only covers kernel 3.2, gives initial insights into the health state. A changelog at functional level and a feature matrix by kernel version, like then one in the Btrfs wiki [9], are completely missing.

Linux originally served as a prototype platform for the implementation of NFSv4.1. As a result, all reasonably recent kernel versions offer the functionality of NFSv4.1. According to the kernel documentation, the implementation of the NFSv4.1 server focuses on the mandatory functions defined by the NFS standard [10].

Compared with NFSv4, NFSv4.1 offers, among other things, sessions, directory delegates, and in particular, parallel access to files stored on multiple servers through pNFS.

Data Collection for Storage: pNFS

If multiple clients try to request and edit data at the same time, the NFS server quickly becomes a bottleneck, especially if the files are distributed over multiple hard disks and storage systems. pNFS seeks to parallelize data access and thus eliminate typical NFS bottlenecks.
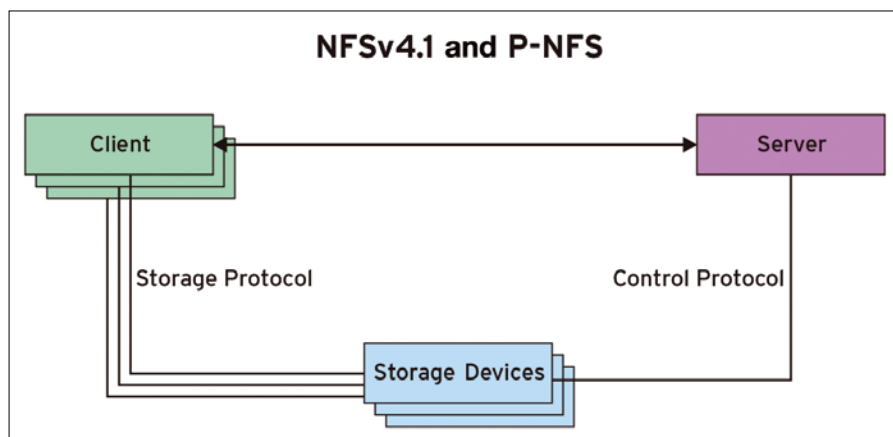
Figure 2: With pNFS, the clients directly access the storage systems.

The first pNFS implementation appeared in 2006 in Linux kernel 2.6.14; it never made it beyond the prototype status. Caution: Quite a few documents on the Internet confusingly refer to this early implementation.

With a current kernel, the NFS server only acts as a metadata server in a similar way to distributed filesystems such as Ceph. In this capacity, it only tells the clients where they can find the requested data. The clients can then optionally request the files directly from the storage systems (Figure 2). pNFS is factorial; the client can also use regular NFSv4 I/O [8].

When a client wants to access the data, it first knocks on the NFS server's door. The server tells the client where to find the data and which protocol it needs for access. This and other meta-information is summarized by the NFS standard in what is known as a layout [11]. Depending on the type of storage, the layout may contain other data. The NFS standard distinguishes four types of layout: File, Block, Object, and Flexi-Layouts.

Only the specification of the file layout is part of the NFSv4.1 standard; the other layouts are defined by separate standards. With a file layout, clients can work directly on individual files that are distributed across multiple servers. An implementation based on the Global File System 2 (GFS2) distributed filesystem [12] does exist under Linux.

The block layout allows access to distributed, block-based devices. According to the kernel documentation [13], the Linux NFS server currently only exports the XFS filesystem via the block layout; this also needs to reside on a distributed memory system – typically an iSCSI array. The filesystem also needs to exist directly on the exported volume; Linux does not allow techniques such as striping or concatenation of volumes as of this writing. The server automatically selects the block layout as long as it supports the filesystem. To make access work, the client needs to build the kernel with the `CONFIG_PNFS_BLOCK` option enabled, run the `blkmapd` daemon from the NFS-utils, and mount the filesystem with the version 4.1 protocol (using `mount -o vers=4.1`).

To avoid data loss, it is essential for the NFS server to block any non-responsive clients. To block non-responsive clients, the server needs a fencing script, the content of which is not documented. But at least the kernel documentation provides a small, uncommented sample script [13].

The new pNFS SCSI Layout by Christoph Hellwig promises to improve the situation; the layout works especially in the context of the XFS filesystem. Clients can directly access the SCSI LUNs using the SCSI layout. The file server currently needs XFS, and striping and concatenation are not allowed. The server automatically enables support for SCSI-layout if:

- the kernel is built with the `CONFIG_NFSD_SCSI` option
- you exported the filesystem with the `pnfs` parameter
- the SCSI device can handle persistent reservations.

On the client side, the conditions are the same as for a block layout.

In the case of the object layout, access is usually via T10 Object-based Storage Device Commands (OSD) and thus relies on specific SCSI commands. On Linux, you will find an implementation based on the EXOFS object filesystem with RAID 0 striping, RAID-1 mirroring, and RAID 5.

The flexible file layout, which is typically abbreviated Flexfiles or Flexi-Layout, is fairly new. Flexible file is designed to reduce the communication with the metadata server [14]. A first implementation made its way into the kernel a year ago.

The reading material is also poor for pNFS: Admins need to rummage through the texts in the kernel documentation [15].

Dual-Protocol File Server

Setting up a file server for Linux and Unix clients using NFS, or a Samba server for Windows clients, is something that virtually every admin can handle. But if you have to support a heterogeneous landscape of users with a single file server, you will automatically be tempted to set up shares with both NFS and Samba. On the network-side, this is no problem. The actual problems for a Janus-headed file server result from the different locking types in Windows and Linux and impending inconsistencies if clients cache data. Another difficulty: NFS and CIFS organize their access control lists differently, which previously meant that only a subset of POSIX 1003.1e ACLs and Windows ACLs remained as the smallest common denominator.

NFSv4 significantly reduced the difficulty of dual-protocol operation by more closely reflecting features of Windows filesystems, and of CIFS in particular, structurally, with mandatory locking, share reservations, and delegations. NFSv4 also transmits the user and group information as UTF-8 plain text and no longer as a numeric identifier, as in NFSv3. The idmap service then converts usernames and groups to the file-server identifiers, which makes cooperating with Samba easier.

The whole thing is standardized: RFC 3010, from December 2000, established a new ACL system that is tailored to NFSv4. For example, ZFS supports ACLs exclusively according to this standard.

Greg Banks from the Cyrus IMAP project presented the Richacls project [16] at the Ottawa Linux Symposium 2010 – an implementation of NFSv4 ACLs in the Linux kernel with advanced file

masking – that superimposes a consistent file rights model on all remote filesystem protocols and accesses both servers and clients. Greg Banks's co-worker on this was Suse employee Andreas Grünbacher, who is with Red Hat today. Banks manages the active Richacls project, which includes kernel code and also patches for Samba, NFS, and XFS.

NFSv4 supports *named attributes* – an additional byte stream that the NFS stores along with the actual file [17]. Windows supports a similar concept called *alternate datastreams* for its NTFS filesystem. Named attributes, by the way, are not related to the extended attributes of the ext filesystem.

Multihosts and Zero Copy

NFSv4 shares resources via a pseudo-filesystem (virtual root filesystem) so that the admin can define arbitrary share paths. Clients mount all shares with a single `mount` command, which only references the NFS server. As of late, an NFSv4.1 client can also specify multiple hostnames when mounting, which promises to relieve the administrative burden of dealing with frequently changing server names.

To increase the speed, clients can establish multiple connections with the NFS server. NFS groups multiple connections in a session in what is known as session trunking [18]. There are other performance-enhancing measures in the current NFS. In case of a copy operation, the client requests the data to be copied and then copies it back to another target on the server. This method is obviously inefficient.

But with zero copy, the server itself can perform the operation, which requires only a copy-on-write copy in the simplest case. Users can create these copies manually on Btrfs and XFS systems with the `cp --reflink` command. The filesystems only allocate additional storage in case of writes to the copy.

Samba supports zero copy with an IOCTL. And there's a technically very charming variant of this that outsources the virtual copy action to Btrfs [19]. NFSv4.2 offers another corresponding approach, but the patches by Anna Schumaker, who works for NetApp, are still under review [20].

Conclusions

The de facto demise of Sun Microsystems has heavily affected the NFS project. On the one hand the development – at least in terms of version numbers – is progressing very slowly. It is difficult to accurately forecast whether there will ever be an NFSv5 release. On the other hand, the missing or outdated documentation slows down any user who wants to try out new features: It is difficult for any admin to know what functionality exists in the NFS implementation packaged with their choice of Linux distribution.

Although some features work seamlessly with the current kernel and the NFS-utils, admins are forced to perform extensive research for many optional features – and to gain experience with a test setup. One complicating factor is that not every Linux distribution enables all the NFS innovations in the kernel or provides the latest NFS-utils.

The specification of the current NFSv4.1 dates back to the year 2010. Although work on NFSv4.2 began as early as 2011, the corresponding specification [21] still only exists as a design today. Among other things, version 4.2 can deal more efficiently with sparse files, named attributes, and zero copy. Newer kernels already support some functions from the NFSv4.2 specification.

These welcome developments are in risk of disappearing, however, because the current home of NFS, IETF, does little to promote NFS as a potential competitor for Samba. ▪▪▪

▮ AUTHOR

Martin Steigerwald works as a coach and performance specialist at Teamix GmbH in Nuremberg. His work focuses on cross-distribution Linux training. Currently, he is developing new training on infrastructure services in Linux, also including NFS. Privately, he is a fan of the contemporary dance style Contact Improvisation.

▮ INFO

[1] IETF, "Network File System (NFS) Version 4 Minor Version 1 Protocol": *https://tools.ietf.org/html/rfc5661*

[2] NFS project site: *http://nfs.sourceforge.net*

[3] nfs-utils Patches: *http://git.linux-nfs.org/?p=steved/nfs-utils.git;a=summary*

[4] Linux NFS wiki: *http://linux-nfs.org/wiki/*

[5] NetApp: *http://www.netapp.com*

[6] pNFS: *http://www.pnfs.com*

[7] Linux NFS mailing list: *majordomo@vger.kernel.org?body=subscribe linux-nfs*

[8] Webinar "pNFS – Are we there yet?": *http://www.panasas.com/news/webinars/pnfs-are-we-there-yet*

[9] Btrfs documentation: *https://btrfs.wiki.kernel.org/index.php/Changelog*

[10] Development status of individual NFSv4 features in the Linux kernel: *https://www.kernel.org/doc/Documentation/filesystems/nfs/nfs41-server.txt*

[11] Layout types in pNFS: *https://tools.ietf.org/html/draft-ietf-nfsv4-layout-types-04*

[12] pNFS file layout implementation: *http://wiki.linux-nfs.org/wiki/index.php/PNFS_server_projects*

[13] Kernel documentation for the pNFS block server: *https://www.kernel.org/doc/Documentation/filesystems/nfs/pnfs-block-server.txt*

[14] Internet draft "Parallel NFS (pNFS) Flexible File Layout": *https://tools.ietf.org/html/draft-ietf-nfsv4-layout-types-04*

[15] NFS kernel documentation: *https://www.kernel.org/doc/Documentation/filesystems/nfs/*

[16] Andreas Grünbacher, "Richacls – Native NFSv4 ACLs on Linux": *http://www.bestbits.at/richacl/*

[17] Named attributes: *http://tools.ietf.org/html/rfc7530#section-5.3*

[18] Session trunking in NFSv4.1: *https://tools.ietf.org/html/rfc5661#section-2.10*

[19] Server-side copy for Samba: *https://wiki.samba.org/index.php/Server-Side_Copy*

[20] Anna Schumaker and others, "NFSv4. 2: Add support for the COPY operation": *http://thread.gmane.org/gmane.linux.nfs/77304*

[21] Internet draft, "NFS Version 4 Minor Version 2": *https://tools.ietf.org/html/draft-ietf-nfsv4-minorversion2-41*

How they test: Papeeria, ShareLaTeX, and Overleaf

Ooh La LaTeX

Authors who publish documents or books with LaTeX can tap into online services using modern browsers, on both desktop and mobile devices, and as a collaborative tool. *By Harald Jele*

Users who have come to understand and appreciate the advantages of a typesetting program like LaTeX [1] prefer to use it exclusively when drafting text. These same users also want it to be available for everyday use on all of their devices. In fact, various possibilities do exist for installing the program on tablets and the like. However, the installation process comes with numerous problems. Consequently, even experienced users are not always able to complete installation successfully without apps like TeX Writer [2] or VerbTeX [3] already running on the target platform.

Therefore, the idea of using LaTeX in an online browser is appealing, because it is a simple and elegant solution that gets around the problems encountered when installing other applications. Addi-

tionally, online access makes it possible to work in groups and take advantage of versioning. In this article, I examine three candidates – Papeeria [4], ShareLaTeX [5], and Overleaf [6] – to see whether online services offer faultless capabilities or whether users are forced to accept restrictions so painful that they forego using the services altogether.

Building, distributing, and maintaining this type of online service is a complicated endeavor. Not everybody is up to the task, as is evident from the failures experienced by such services as Fidus Writer [7], FlyLaTeX [8], SpanDeX [9], LaTeX Lab [10], and MonkeyTeX.

How They Work
The approaches taken by the online services tested here share striking similarities:
- They install one or more releases of TeX Live [11] as the standard LaTeX on a Linux server.
- The files to be edited land in the user's document directory on the server, and the service automatically saves them there regularly.
- Users edit the files remotely in their browser via an online editor; mean-

while, the editor allows multiple users to share a file.
- An online PDF viewer displays the LaTeX document that has been compiled on the server. The compiling process is the same as the one LaTeX uses on a local installation.

This seemingly simple concept can only be implemented with the involvement of a wide range of technologies. The online editors are mostly implemented as asynchronous JavaScript applications.

The Competition
The test data came from two existing LaTeX projects that were uploaded and compiled online, with the results checked thoroughly. In this article, I looked at a project shared between coworkers, as well as a project in which work was performed simultaneously by multiple users on the same file.

Those who want to create new projects will find that the services typically offer a plethora of templates. For the most part, they are applicable to dissertations and articles for scientific journals. The number of file templates should not be the criterion you use to decide for or against a

AUTHOR

Dr. Harald Jele works at the University in Klagenfurt, Austria. He became acquainted with LaTeX in 1993 after switching to a Linux server and desktop. Since then, he prefers not to be without either.

Lead Image © Zoya Fedorova, 123RF.com

particular service, though. Because all LaTeX services behave like a local installation, it suffices to put any random template in the document directory. LaTeX finds the file and integrates it. The same goes for LaTeX packages that you want to load during compilation. If they are not preinstalled, you only need to get them via the Internet and move them to the document directory.

LaTeX can handle any number of text types in addition to scientific text, including everything from Advent calendars to greeting cards. However, the people interested in these types of documents are probably not interested in the online services I talk about in this article. One as yet unresolved issue is that none of the templates come with information on the LaTeX compile settings. This goes for all of the services described here. Presumably, most users compile with pdflatex, whereas a few turn to XeLaTeX to get working PDF files.

Additionally, all three services offer free access to a more restricted version than the fee-based choice in terms of time period for use or the number of available functions.

Papeeria

Although Papeeria [4] was created by a Russian company, the LaTeX user is barely aware of it because most of the

pages are in English. When using free testing access to the program, it is important to remember that the platform makes all saved documents publicly accessible after the first project. A fee-based account costs $5 dollars per month.

To begin, you upload an existing LaTeX project as a ZIP archive, and the service decompresses and puts the files in your document directory. The standard setting uses TeX Live 2015. If problems occur, you can turn to a 2013 version to compile.

The developers have not corrected all of the installation errors. During testing, I found that installation of the EB Garamond font was faulty. Once notified, Papeeria responded promptly via email that three full-time programmers had received an SMS alert. However, I was still able to reproduce the bug two months later; thus, it would seem that bug reports are ignored in spite of intensive communication efforts.

Figure 1 shows the very straightforward interface. Those familiar with LaTeX will be able to navigate easily. The four frames display the project files, LaTeX code, PDF presentation, and error messages and warnings.

Pretty Tired

Unfortunately, users who chooses the free test version are informed that their jobs

are queued to compile behind all of the paying customers. The right-hand frame in Figure 1 shows the dialog window.

In most cases, this downgrade also meant that the website displayed the cached or erroneous modules instead of those most recently updated. Therefore, free test access does not actually work for meaningful testing. The tester should request access that does not have these restrictions; otherwise, you will check your well-written code in vain for errors. The situation is similar with embedded graphics, which are often not depicted by Papeeria in test mode, and a warning failed to appear that would explain what caused the behavior.

It was possible to compile the two uploaded LaTeX projects with both the TeX Live 2015 and 2013 releases. The results for the 2013 version added empty pages, and none of the available settings corrected this issue.

If you compile frequently to do things like code-checking, you will encounter yet another problem with Papeeria. The embedded scripts demand so much performance from the browser that it offers to stop the running scripts.

On a positive note, Papeeria has done a good job of implementing both the visual display and the menubar. The menu provides quick access to essential tasks and makes the service easy to operate,
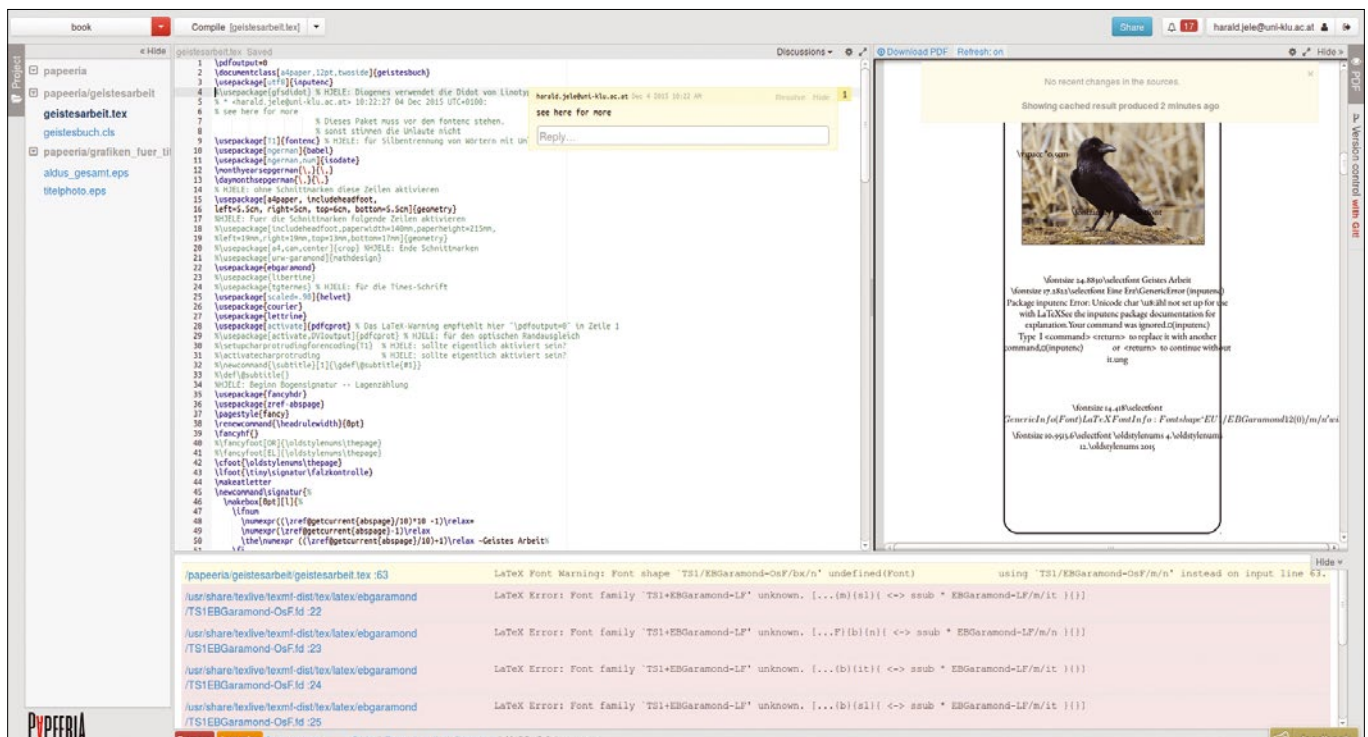
Figure 1: The LaTeX online platform Papeeria divides the work area into four frames and offers TeX Live 2013 and 2015.

so it is not necessary constantly to switch between various screen pages. Because the online editor responds correctly to the usual shortcuts, you do not have to learn new ones.

The associated spell checker does not know many German language terms (my language), which gave me the impression during testing that language sorting does not work properly and would not be suitable for professional use.

The editor only supports UTF-8 character encoding and displays text in Western ISO Latin 1 incorrectly. However, LaTeX reacts well during compilation, as long as you do not make any changes to the text that would serve to create faulty character encoding. Even then, the editor displays the compiled Latin 1 texts incorrectly, albeit not each and every time (Figure 2).

DVI, AUX, and LOG (device-independent binary, auxiliary, debugging) files created during compilation are not accessible; you can only download the PDF files from the server. Papeeria offers the following options for compiling: pdflatex, XeLaTeX, Legacy TeX Live 2013, and PGF 3.0.

Divide and Plot

If you want to share text with other users, you click the *Share* button at the upper right. The platform then sends an email

message to the target users, who need to confirm. The user interface is silent during this procedure, and the need for action is only revealed when your coworkers, perhaps accidentally, take a look at their inboxes. Papeeria applies color highlights to each change and marking made in the text by the sharing recipients. The program augments the highlighted changes and markings with the email address of the user who made them.

In addition to LaTeX, the platform offers gnuplot [12], which can create impressive graphics from existing data. However, this utility is only available with a fee-based account. The free version includes a simple plot builder.

Thanks to Papeeria's session management, it is possible for a user to work on multiple projects simultaneously, supporting team effectiveness. If this option is missing, you have to log out of a project constantly to jump into another one before helping out. Then the logout-login procedure has to be repeated to get back into the other project.

Support (e.g., for operation of the interface with embedded programs such as the editor, PDF viewer, file browser, etc.) is not well integrated at all, including interaction with LaTeX itself. Contacting the developers is possible through various services like Twitter, Google Plus, and TeX-LaTeX Stack Exchange [13].

However, it takes time to find the right people and the proper communication channel.

Paying customers can operate a maximum of 10 private projects and synchronize them with public and private Git repositories. Additionally, Papeeria offers regular snapshots for back ups.

ShareLaTeX

Most of the developers working on the second online service tested, ShareLaTeX [5], are British [14]. The start page evaluates the language of the browser and indicates that your interface will display in your language, if supported, which you select with the flag symbol located at the lower edge of the page. However, ShareLaTeX documentation and support are only offered in English.

Once you log into ShareLaTeX, the first thing to appear is an overview of existing projects. New projects are set up in the overview by loading existing LaTeX files as a ZIP archive into the corresponding document directory or by selecting a template for starting a new project. Alternatively, you can import a project from GitHub. The overview helps keep you organized if you are working on multiple projects simultaneously and want to participate in more.

Also in the overview is an easily accessible and convenient logout button
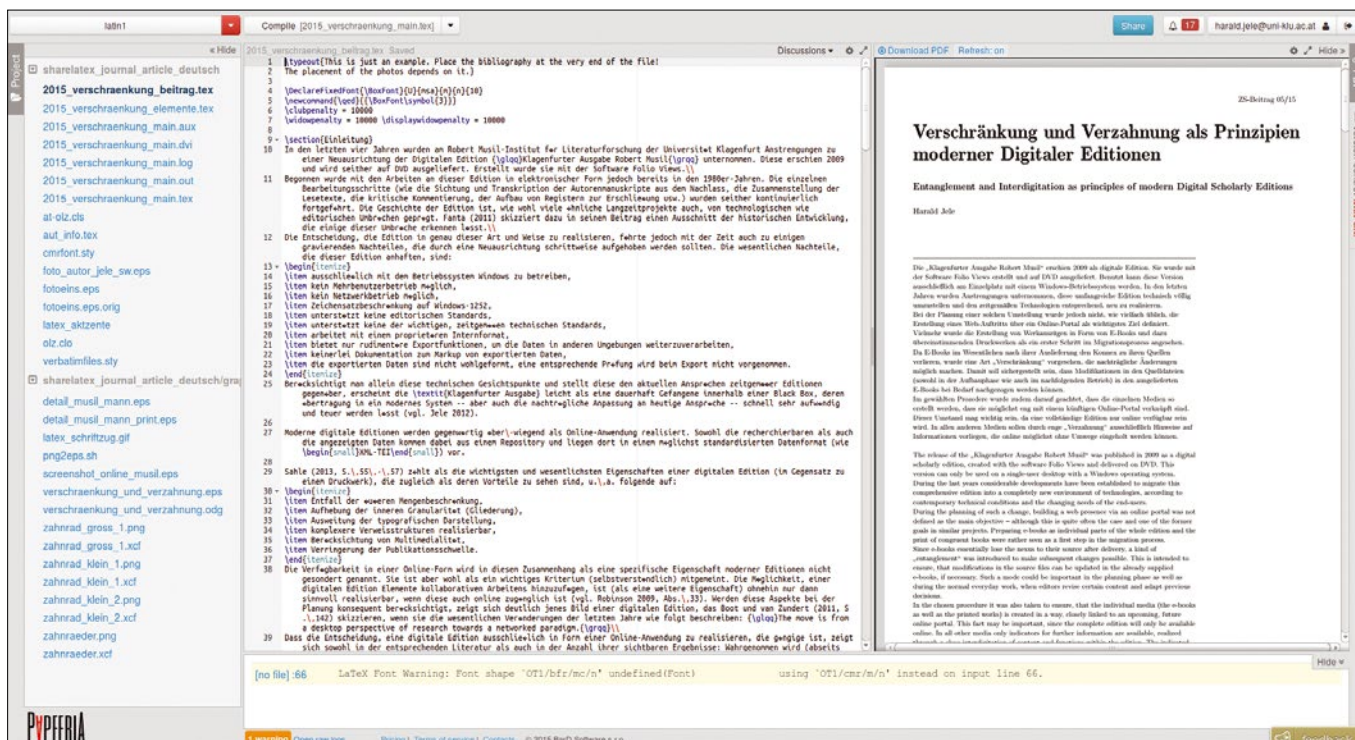
Figure 2: Even an erroneously displayed text encoding can give correct results. By default, Papeeria is set for UTF-8.

for those working on multiple projects. The button is only available on this page and is not found elsewhere. The user should keep in mind that only one project can be open per browser session. The session administration currently implemented does not handle multiple projects in various browser windows or tabs. Therefore, you would have to install a number of browsers to work in parallel on projects. When it comes to teamwork, this could be particularly disadvantageous.

When you open a ShareLaTeX project from the project directory, another view opens for editing the files. Once the online platform recognizes a `main` file, it will compile the file and display the PDF generated in a frame next to the editing area. Usually this occurs surprisingly fast. Even a very large piece of work with 300 pages appears within just a few seconds, thus giving the impression that you are sitting in front of a local installation.

The screen display is divided into coherent frames, each of which is devoted to a particular task (Figure 3). To the far left is a simple file tree. This is where the files you have uploaded appear. The support and logfiles remain hidden; however, they become visible when you download the document directory with the DVI files.

General settings applicable to the current project are located above the file browser. Settings include options for compiling, language selection for spell checking, and a few display settings. The compiling utilities offered by the service include pdflatex, LaTeX, XeLaTeX, and LuaLaTeX.

Leaping

The online editor and the PDF viewer lie in the next two frames to the right. You can jump in both directions with an arrow key, switching from a PDF document to the relevant place in the editor, and vice versa. This feature proves to be extremely helpful when working with LaTeX. Buttons for compiling, viewing the log, and access to other output files are placed above the PDF viewer.

The frame to the far right provides an area for project-related conversations. The design is generally modest in a positive sense, although the unified menubars in Papeeria occasionally organize work somewhat more elegantly.

The frame division proves to be somewhat inconvenient when you are working in the editing layer if you want to look up documentation or support functions, which are only accessible at the project layer. The developers are not currently planning to add context-sensitive support.

Editing functions are limited to the essentials, which is perfectly adequate for speedy and smooth work with LaTeX.

Automatic code completion is a well-implemented feature and is turned on and off separately for each project. An interesting addition for those who have only a little experience with LaTeX would be to make the LaTeX code invisible so that only the content could be seen. Unfortunately, the editor does not offer this feature.

The editor supports files with UTF-8 encoding exclusively, which means projects that have been in progress for decades cannot be continued when changing over to ShareLaTeX. According to the developers, this aspect of LaTeX is rarely criticized; therefore, it will probably not see any changes in the near future.

The LaTeX version installed is TeX Live 2014, which compiled all of the prepared documents without any errors. However, I could not figure out why some of the texts did not produce a valid PDF with pdflatex, even though the DVI file generated could be converted without problem into error-free PS and PDF files. Once I compiled the text again, I was always successful. The error appeared only rarely during testing and was possibly the result of short periods of high server load.

TeX Live behaves like a local installation. If you move LaTeX packages and style files not made available by the provider into the document directory, LaTeX finds them during compilation. If
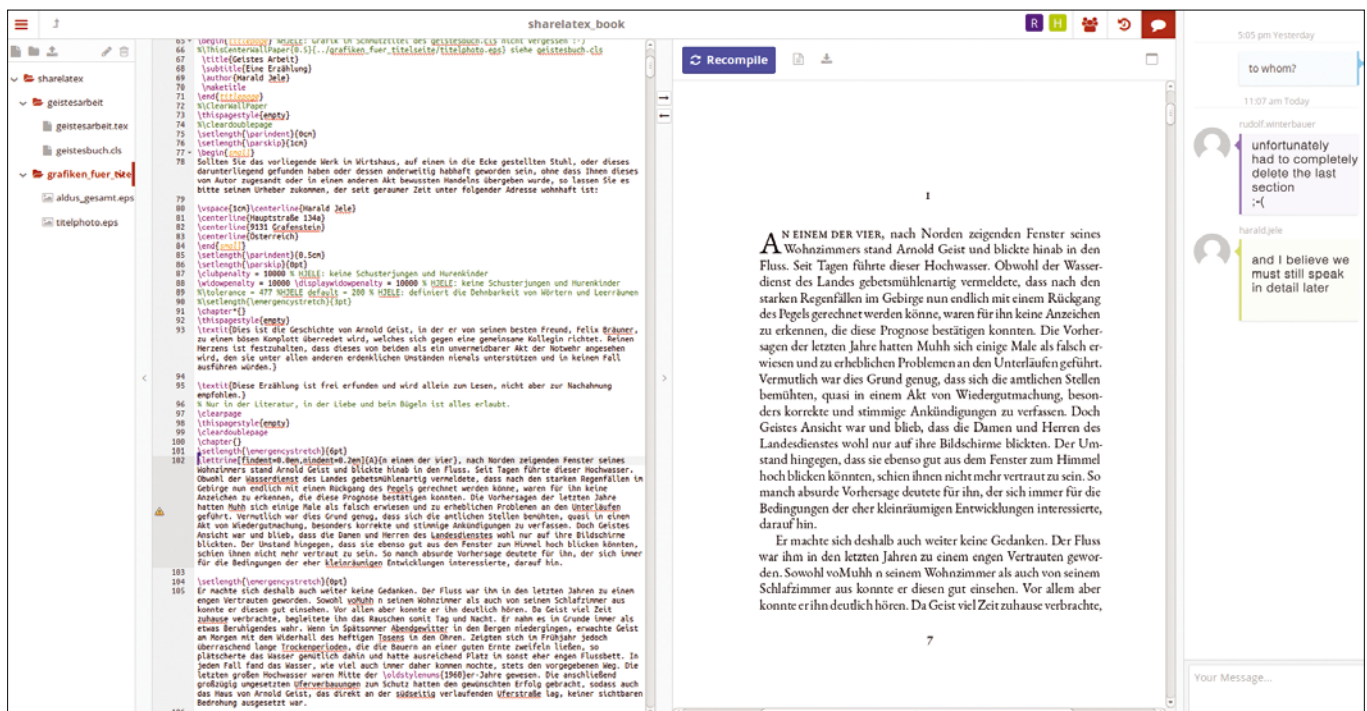
Figure 3: ShareLaTeX divides the screen into frames. Project files appear to the left, LaTeX code is in the middle, and the results plus comments are to the right.

requested packages are missing, the ShareLaTeX team installs most of them when asked to do so.

Very Inviting

The ShareLaTeX user can make a project available at the editing layer, either with write permission or as read only. The share settings have been well designed. Invited collaborators do not have to do anything to accept the invitation. ShareLaTeX lists the most recent changes to the open file right next to the share list (Figure 4). This file history, or *Recent Changes,* lets you undo changes or recreate them. The history also shows which user has made which change. The service synchronizes the project files with other services and local installations via Dropbox and GitHub.

ShareLaTeX has an impressive number of templates for creating new text files. Unfortunately they are not tagged, nor are they very well sorted. For example, if you go to the *Thesis* category, you will find 185 templates for use with undergraduate theses, dissertations, and undergraduate and graduate papers. The program advises you as to which template fulfills requirements set by a particular university. However, the program is silent on the topic of how the template is built and the settings used for compiling it. You will also look in vain for a filter

that restricts the view to templates that are compilable with XeLaTeX. Templates for use outside of the scientific disciplines are in short supply in the ShareLaTeX directory.

ShareLaTeX costs $15 per month for 10 coworkers and $30 per month for an unlimited number of coworkers. The service also incorporates a fee-based editing service, Enago [15], which lets you purchase text-related services ranging from proofreading to text editing.

Large portions of the project are freely available in the source code. This means that users can look at the source code if they are thinking of building a similar service or if they want to get a glimpse behind the scenes [16]. ShareLaTeX also offers a service named DataJoy [17] for the statistical computing language R and for working with Python, although the service is not accepting new accounts and will be shut down completely on January 2, 2017.

The developers were easy to communicate with and information exchange went quickly. The web pages clearly state who is responsible for what.

Overleaf

The online service Overleaf [6], previously known as WriteLaTeX, was started in 2011. The goal of the founder was to support all users who wanted to compose

their text with LaTeX and exchange their work with teams at the same time. The thought was that an online service would allow users to focus their attention on content instead of spending so much time dealing with installation issues.

Once logged in, you land directly in the project directory as you would with other services. The directory contains the usual project-related functions (e.g., create new file, rename, delete, archive, copy). Additionally, you can *clone* LaTeX projects to GitHub from here. All of the other functions are found on the editing layer.

An extensive set of templates are available for setting up a new project, and they come with comments and are tagged with keywords. All users, not just scholars, can profit by using Overleaf templates. Some templates can be used for everyday office activities, creating posters, sending visually pleasing business correspondence, or designing a calendar for annual vacation planning.

Figure 5 shows the clean display in editing mode, which is only available in English. The screen looks tidy and well put together thanks to the organization of the frames. Compared with ShareLaTeX, the horizontal arrangement of frames has a simple and easy-to-understand menu structure that is much more appealing.

The left frame is used for file administration and can be hidden and displayed
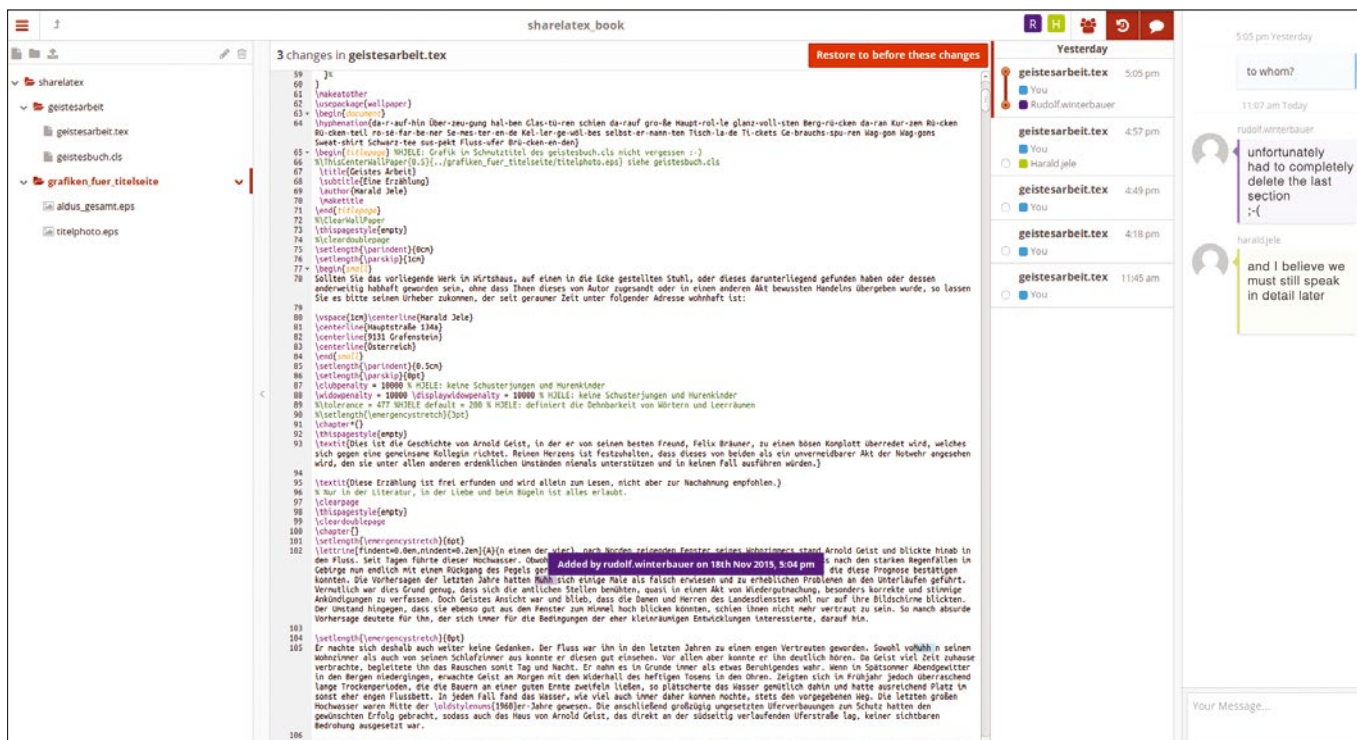
Figure 4: The file history shows the details of any changes made to the document.

by clicking the *Project* button that sits above. This frame also contains commands for downloading the project files from the server (*Download as ZIP*) or for depositing them in a Dropbox account. Google Drive users can also upload files. Overleaf hides DVI, AUX, and LOG files in this structure. However, the files are found in the downloadable ZIP archive of each project.

Like ShareLaTeX, the Overleaf division of functions is somewhat inconsistent. It is unclear which belong to the project layer and which to the editing layer, although you can easily become familiar with this situation in both services. The nearby buttons *Versions* and *Share* are self-explanatory: the *Versions* button lets you compare multiple versions of a file in frames that sit side by side, and *Share* generates links for write permission or read-only access to the project files to be shared.

When sensitive content is involved, you should exercise caution, because everyone who receives a link has user rights. The fee-based version, however, has extended access control. During testing, I was surprised to see links that provide access to projects that had already been deleted.

Behind the *Publish* button are a series of repositories of scientific literature. If you want to publish your project here, you can set it up easily.

Linux Ahoy!

With Overleaf, you have project-specific and overlapping settings under the *Settings* icon on the right side of the screen. A pleasant discovery is that the default settings for the editor can be adapted to Vim and Emac. This should make Linux users happy, especially because numerous editor themes are provided.

The second frame containing the editor can be hidden or displayed. Its menu structure offers various common functions important in dealing with LaTeX, like keyboard shortcuts.

The *Rich Text* display mode should be interesting in the context of team work. Overleaf has put numerous control commands and LaTeX code sequences there and displays the results they generate. This is an experimental feature, but it should nonetheless be helpful for those who are put off by the thought of text coding.

Overleaf does not display comments in its own frame and instead embeds them in the text. Short text is therefore difficult to administer and read. The conversation structure in ShareLaTeX is easier to use. Moreover, the chat information in Overleaf gets lost when a project is transferred from one colleague to another.

The editor displays the UTF-8 character encoding and other international character encodings correctly. Unfortunately, a valid PDF file did not appear

when the test file was compiled in ISO Latin 1 text encoding. Strangely, Overleaf repeatedly failed to generate PDFs even though the DVI files that had been created appeared to be free of errors. These files also could be converted into valid and display-ready PDF files without problem. Apparently, the Overleaf programmers are taking the LaTeX errors and warnings too seriously. As a result, the platform will occasionally not display functioning PDF files.

The current version of Overleaf uses the TeX Live 2015 distribution. This behavior, as noted with the other two services, is conspicuously inconspicuous.

When you encounter frequent coding problems in LaTeX, it is a good idea to read the help files and logfiles created during compilation. Papeeria and ShareLaTeX make these available online. With Overleaf, you will need to evaluate the data locally, which is cumbersome on a notebook, extremely difficult on mobile devices, and therefore ultimately one of the major shortcomings of the service.

Helpful

Both LaTeX and Overleaf help is available from a button located on the right side of the screen in the horizontal bar. Thoughtfully organized, a search function quickly takes you to important information. The remainder of the LaTeX
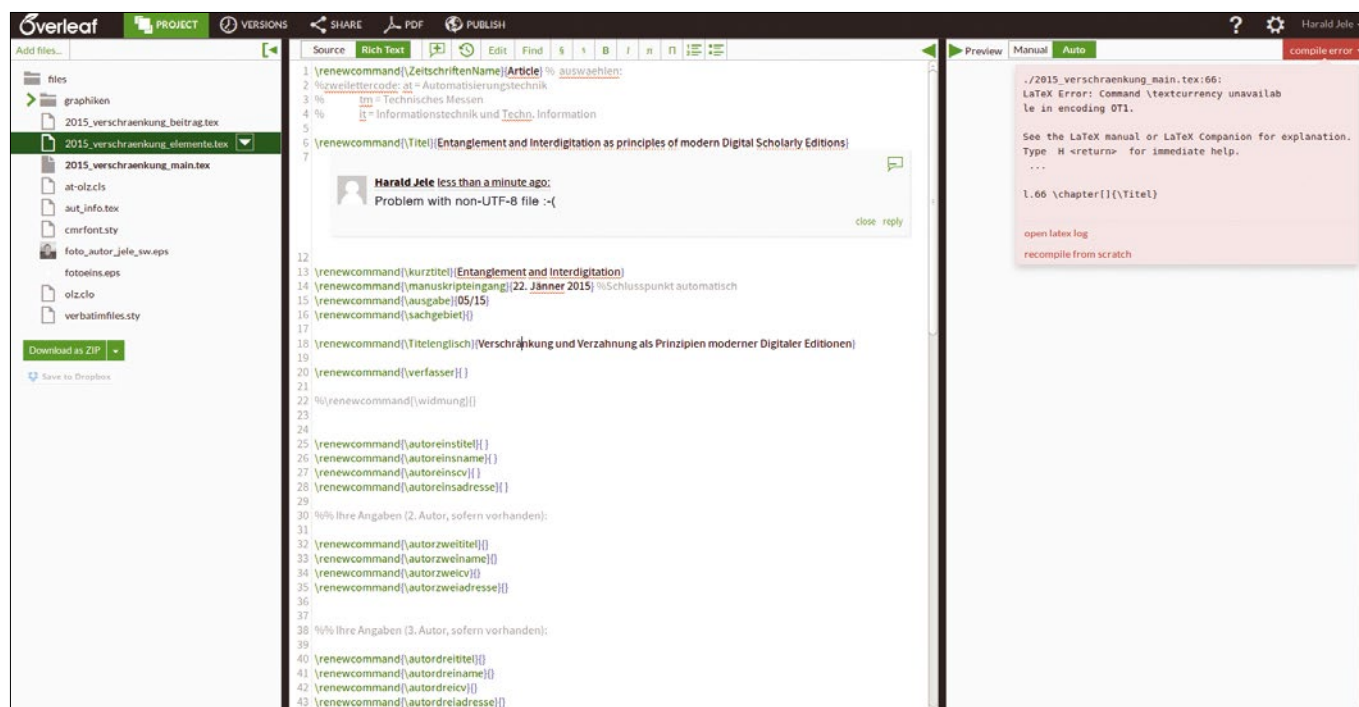
Figure 5: Editing mode in Overleaf, which only comes in English. LaTeX displays an error message when you try to compile a file with ISO Latin 1 encoding.

body of knowledge is typically found using your favorite search engine.

Session management in Overleaf also prevents you from participating in multiple projects at the same time unless you log out and back in again. Sadly, you often gain important insight just after logging out. LaTeX users who administer their bibliography and cited sources in clear text with Zotero [18] or Mendeley [19] find that Overleaf accepts data from both services.

The free version permits an unlimited number of projects and collaborators provided there are no more than 60 files per project. The fee-based versions cost $8 and $12 per month for 10GB, and no more than 500 files, and 20GB, and no more than 1,000 files, of storage space, respectively.

Communication with the Overleaf developers via the contact form on the website and by email worked perfectly. Their answers were helpful and delivered promptly. As it turned out, though, we had the fewest number of questions with Overleaf.

Conclusion

If you want to use LaTeX anytime and anywhere, the three online solutions presented here are eminently practical. Authors who lack funds are better advised not to use the free version of Papeeria, because it has significant disadvantages

not found in its fee-based counterparts that impede productive work.

All three providers expend considerable effort to create a frictionless environment for teamwork, and they have succeeded. Two of the providers, ShareLaTeX and Overleaf, are deficient in session management by preventing users who want, or need, to work on multiple projects from achieving effective teamwork. Papeeria shines here because its developers have taken this particular scenario into account.

To derive reasonable use from the services, it suffices to have basic LaTeX knowledge and a modern browser. Large display screens have a positive effect on productivity. Users with mobile devices roughly the size of a smart phone should probably rely on a locally installed app like TeX Writer or VerbTeX. Both of these do much better with display screens ranging from 4 to 6 inches diagonally.

The load and compile times of the online services are short and definitely competitive with those the average notebook achieves. This is equally true for large documents with complex document structure.

The big advantage to online services like those discussed here is that they let you concentrate on content instead of installation. Therefore, it is possible to get started quickly in an educational environment and in a productive work environment. It

doesn't matter whether you are working alone or on a team, everything sets up and works quickly. Team members can help one another by letting knowledgeable participants deal with more complex LaTeX properties, so the remainder of the team is not slowed down or does not become discouraged. This reason alone is a big plus.

An online LaTeX editor cannot be used as a substitute for a conventional text editing system. The display in rich text mode helps bring less confident colleagues on board large TeX projects. Although the services are intended for scholars, they can also be helpful in publishing other material. For example, publishers can offer their authors a professional encoding system that helps avoid unwieldy manuscripts.

When comparing the services, you will quickly determine that each comes with both clear advantages and at least one shortcoming. Table 1 helps acquaint you with these characteristics. ■■■

■ INFO

[1] LaTeX: *https://www.latex-project.org*

[2] TeX Writer: *http://www.texwriterapp.com*

[3] VerbTeX: *https://www.verbosus.com*

[4] Papeeria: *https://papeeria.com*

[5] ShareLaTeX: *https://www.shareLatex.com*

[6] Overleaf: *https://www.overleaf.com*

[7] Fidus Writer: *http://www.fiduswriter.org*

[8] FlyLatex: *https://github.com/alabid/flyLatex*

[9] SpanDeX: *http://texwelt.de/blog/spandex-schliesst/* (in German)

[10] LaTeX Lab: *https://code.google.com/archive/p/latex-lab/*

[11] TeX Live: *http://www.tug.org/texlive/*

[12] gnuplot: *http://www.gnuplot.info*

[13] TeX-LaTeX Stack Exchange: *http://tex.stackexchange.com*

[14] ShareLaTeX developer: *https://www.shareLatex.com/about*

[15] Enago: *http://www.enago.de*

[16] Source code for ShareLaTeX: *https://github.com/shareLatex/*

[17] DataJoy: *https://www.getdatajoy.com*

[18] Zotero: *https://www.zotero.org*

[19] Mendeley: *https://www.mendeley.com*

TABLE 1: Feature Overview

Feature	Papeeria	ShareLaTeX	Overleaf
International character encoding	No	No	Yes
Syntax highlighting	Yes	Yes	Yes
Code completion	Yes	Yes	Yes
Useful editor adaptations	No	No	Yes
Rich text editing and display	No	No	Yes
Multilanguage interface	No	Yes	No
Browser compatibility	Yes	Yes	Yes
Suitable for mobile devices	No	Yes	Yes
Number of functions with free account	Limited	Sufficient	Sufficient
Adjustable print quality	No	No	No
TrueType and OpenType fonts	Yes	Yes	Yes
Management for multiple sessions	Yes	No	No
Synchronization with GitHub and Dropbox	Yes	Yes	Yes
Built-in version administration	No	No	Yes
Quality of the LaTeX installation	Good	Good	Good
Compile settings	Yes	Yes	Yes
Hidden LaTeX coding	Not possible	Not possible	Possible
Team support	Yes	Yes	Yes
Source code available	No	Yes	No
Developers available	No	Yes	Yes

Tool tests on the fast track *By Uwe Vollbracht*

TOOL TIPS

Fstools 20160322

Function: Useful script collection
Source: *http://fex.belwue.de/fstools*
License: GPLv2, Perl Artistic License
Alternatives: DK tools

```
            Terminal - vollbracht@LMLab: ~/extract/tooltips/fstools-0.0/bin    - + ×
vollbracht@LMLab:~/extract/tooltips/fstools-0.0/bin$ ./df3 -v
Filesystem       Type   1K-blocks     Used    Avail Use% Mounted on
/dev/sda1        ext4      24,565    6,382   16,913  28% /
//dcvcon02/daten cifs     143,582   49,077   87,211  37% /home/vollbracht
/exchange
vollbracht@LMLab:~/extract/tooltips/fstools-0.0/bin$ ./portwatch -v 8888
portwatch: 2016-03-24 20:08:23  connect from tcpdesktop.vcondt.de[192.168.250.30
]
Connection established
Firewall free
quit
portwatch: 2016-03-24 20:08:48  tcpdesktop.vcondt.de[192.168.250.30] has closed
the connection
vollbracht@LMLab:~/extract/tooltips/fstools-0.0/bin$ ./lshd

NAME   SIZE FSTYPE LABEL MOUNTPOINT
fd0      4K
sda     25G
├─sda1 24,5G ext4        /
├─sda2   1K
└─sda5 510M swap         [SWAP]
vollbracht@LMLab:~/extract/tooltips/fstools-0.0/bin$
```

Linux has a well-stocked toolbox with command-line tools for every imaginable scenario. Many admins combine them, write extensions, and create their own scripts for repetitive tasks. This is also how Framstag's Shell Tools (Fstools) arose, which programmer and admin Uli Horlacher compiled over the past two decades.

More than 200 shell and Perl scripts for various applications are included – from converting between number systems, file formats or fonts, to procmail extensions and statistical analysis. Other scripts determine the absolute path of a file and generate snapshots of files, if the filesystem supports this functionality. Alternatives to known command-line tools are also included.

The project page contains a complete overview of all the tools and brief function descriptions. Note that man pages do not exist, and the scripts contain hardly any comments. There is a rudimentary online help, however, via -h.

★★★☆☆ The Fstools contain interesting additions to the common shell tools. In terms of documentation, the developer is reticent – a look at the code can help and possibly provide inspiration for your own scripts. ∎∎∎

SUP 1.1

Function: Working with root privileges
Source: *https://git.devuan.org/jaromil/sup*
License: LGPLv3
Alternatives: Sudo, Su

```
            Terminal - vollbracht@LMLab: ~              - + ×
sup ifconfig called by vollbracht(1000) gid(1000)
vollbracht@LMLab:~/extract/tooltips/sup$ ip a s enp0s17
2: enp0s17: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UN
KNOWN group default qlen 1000
    link/ether 00:0c:29:ec:ca:4b brd ff:ff:ff:ff:ff:ff
    inet 192.168.250.193/24 brd 192.168.250.255 scope global dynamic enp0s17
       valid_lft 390sec preferred_lft 390sec
    inet 192.168.1.1/24 brd 192.168.1.255 scope global enp0s17:1
       valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:feec:ca4b/64 scope link
       valid_lft forever preferred_lft forever
vollbracht@LMLab:~$ sup -l
List of compiled in authorizations:

User     UID    GID      Command        Forced PATH
vollbracht 1000   -1        whoami      /usr/bin/whoami
vollbracht 1000   -1      ifconfig      /sbin/ifconfig
vollbracht 1000   -1          ls            /bin/ls
vollbracht 1000   -1        wifi      /root/wifi.sh
vollbracht 1000   -1          id              *

Flags: HASH ENFORCE

vollbracht@LMLab:~$
```

Linux admins can make use of a comprehensive system for assigning rights. You can selectively grant access to files or executables users or prevent this access. Using su, sudo, or the setuid bit gives admins three approaches. SUP is a C program that seeks to establish itself as a fourth alternative, and promises more security thanks to checksums accompanied by a compact design.

The archive includes three files: sup.c, sha256.c, and config.h. The latter accepts the admin's rules. You can define here, for every executable, a line with the UID, GID, the full path of the command, and optionally the checksum of the binary. Users who access a command in this way append it to a call to sup.

The configuration is thus handled in the source code, meaning system administrators need to recompile SUP after each adjustment. This is quite cumbersome, because you need to call make after each change.

★★☆☆☆ The idea is interesting, especially the extension for checking the checksums. However, it is quite time consuming to recompile the tool after every update. It is therefore not suitable for large-scale infrastructures. ∎∎∎

MuPDF 1.8

Function: Simple PDF Viewer
Source: *http://mupdf.com*
License: AGPLv3
Alternatives: Xpdf, Evince
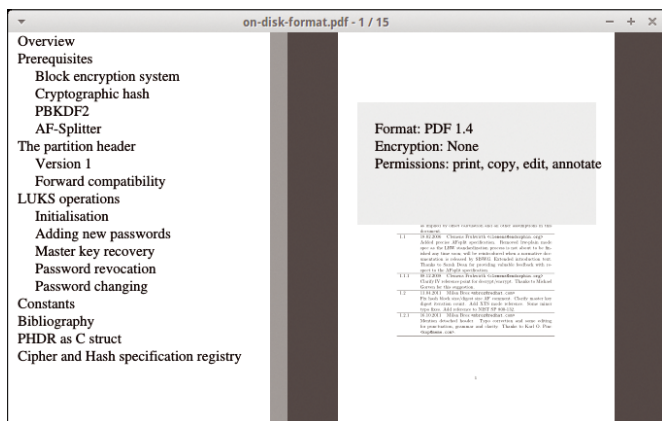
The PDF viewer MuPDF leaves a positive impression in part because of its low resource requirements. It cuts a fine figure not only on Linux or Windows desktops but also on Android and iOS devices.

On Linux, users call `mupdf` along with the name of the file. Documents with password protection require the `-p` option and password – a less than elegant solution because the password appears in the clear in the shell history. Also, MuPDF has only a few options. Besides the resolution and anti-aliasing, users can only define the font and window size. There is no configuration file, and MuPDF does not support links to internal or external addresses.

This lean approach is continued in the interface. There is no a menu or toolbar. Instead, users control the viewer with the keyboard. This means pressing the H, J, K, and L keys to navigate; pressing F switches to full-screen mode and / starts a search. The man page shows a list of all keyboard shortcuts.

★★★☆☆ MuPDF 1.8 is fast – but users should not expect convenience. Older issues of the tools originally contained rudimentary functions for manipulating PDFs, which are missing in the current version. ∎∎∎

Cryptsetup 1.7.1

Function: Encrypting partitions
Source: *https://gitlab.com/cryptsetup/cryptsetup*
License: GPLv2
Alternatives: LUKS, Dm-crypt
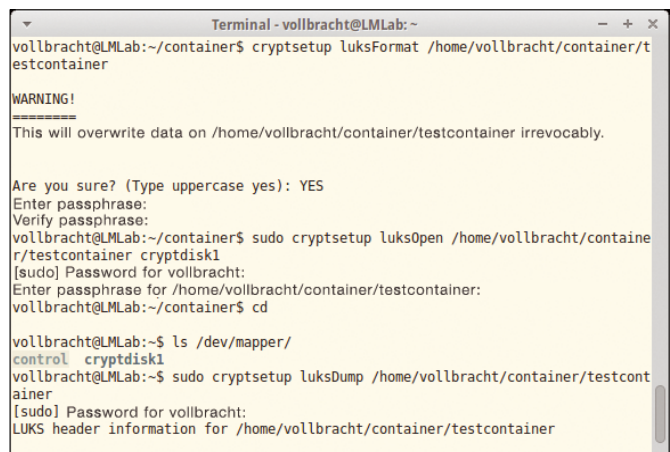
Most solutions for encrypting partitions and containers are very powerful, but they do not exactly impress with ease of use. Cryptsetup offers a comfortable interface to use Loop-AES, in addition to Dm-crypt and LUKS, as well as TrueCrypt with the VeraCrypt extension.

On the project page and in the man page, the developers strongly recommend the use of LUKS. This de facto standard on Linux offers many useful features. For example, users can assign up to eight keys to a device, and revoking keys and passwords is also easy. To encrypt a device, partition, or container with LUKS, you call `cryptsetup luksDump <devicefile>` and then follow the instructions shown by the tool.

If you use a keyfile instead of a password, you can specify it following `--key-file`. Other parameters determine which cryptographic algorithm is used or how often users may enter incorrect passwords. Once the device is encrypted, `cryptsetup open` opens it so that the user can install a filesystem and integrate it into the directory tree. The `status` and `luksDump` commands give insights into the current configuration of the device; `luksHeaderBackup` and `luksHeaderRestore` take care of backing up and restoring the header information.

★★★★★ Cryptsetup is a reliable helper for users who want to encrypt partitions or containers. ∎∎∎

Netjukebox 6.09.2

Function: Manage and stream media
Source: *http://www.netjukebox.nl*
License: GPLv3
Alternatives: Edna, Ampache
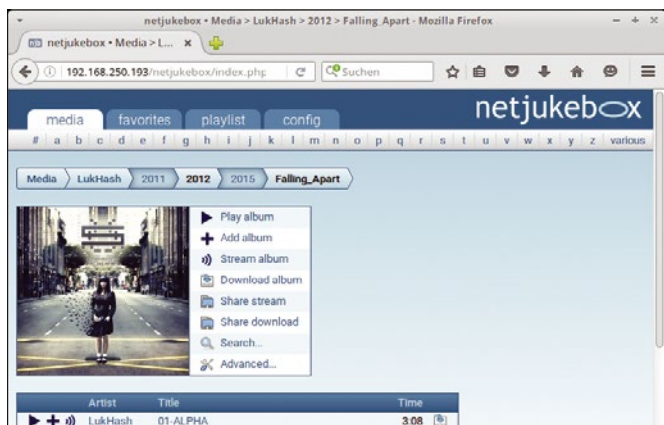
Many users nowadays collect their music in digital form. If you want to listen both locally and on other devices on the network, you need a streaming server such as Netjukebox. The tool cuts a fine figure on LAMP or XAMP servers. The PHP program not only organizes your collection of music but also manages video files, maintaining the metadata in a MySQL database and relying on the music player daemon (MPD) for playback.

According to the instructions, you simply need to unpack the Netjukebox archive located in the document root of the web server and edit the `include/config.inc.php` file in the text editor. Then, enter the media directory and point to the database that you created previously with MySQL on-board tools. You then install the data structure with the `sql/netjukebox_46.sql` script.

The web interface is found at `http://<hostname>/ netjukebox`; the username and password are `admin`. In the `config` area, you can change the administrator password and set up user accounts. Here you will also find a function for updating the database. After that, there is nothing to interrupt your listening pleasure. Netjukebox displays album art, has search and filter functions, and lets you create playlists and download the data to your own computer.

★★★★☆ Netjukebox manages digital music and can be used as a standalone player or streaming server on the network. The configuration is somewhat clumsy – but operation is child's play. ∎∎∎

Pass 1.6.5

Function: Password management for the shell
Source: *https://www.passwordstore.org*
License: GPLv2
Alternatives: Pwd.sh, Steel
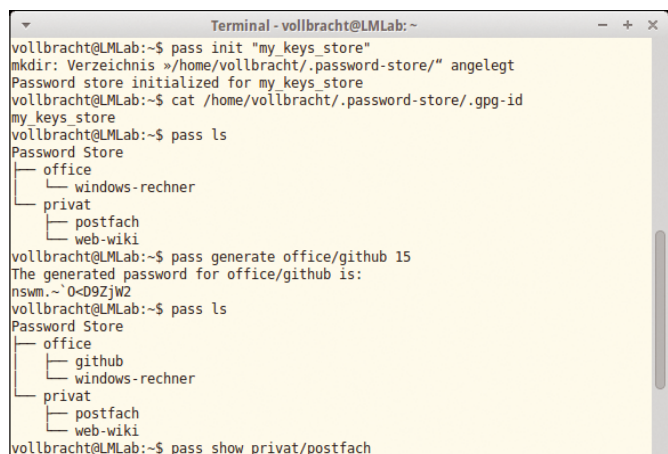
Pass is a powerful password manager for the console that not only handles the process of creating and storing passwords but can also synchronize them with multiple machines with the help of Git. In the background, the shell script relies on tools such as GnuPG (for encrypting passwords), tree (tree view on the console), pwgen (password generator), and Xclip (Clipboard).

To create a new safe, you call `pass init` followed by your GPG key ID. The key information as well as the passwords are stored in the folder `~/.password-store`. Using `pass insert` adds a new password. If you lack ideas for strong passwords, the `pass generate` option gives you access to the support of `pwgen`. Optionally, users can specify the length of the password and a number in the command.

When called with no parameters, `pass` shows you a tree view of all password files that exist; `pass find <term>` allows you to search explicitly for a password. If you need to read a password, `pass` needs this identifier. The `-c` option tells the tool to copy the password directly to the clipboard and keep it there for 45 seconds. The `pass` tool also supports the options `rm`, `mv`, and `cp`. And, `pass git` converts the safe into a Git repository; a look at the man page explains how this works.

★★★★★ Pass does not attempt to reinvent the wheel but instead relies on proven tools and combinations of them. The teamwork with Git is particularly useful. ∎∎∎

RISE HIGHER

EACH ISSUE OF DRUPAL WATCHDOG OFFERS TOOLS, TIPS, AND BEST PRACTICES FOR BETTER DRUPAL WEBSITES.

NOW PUBLISHED 4 TIMES PER YEAR!

Renew or subscribe now!

SUBSCRIPTIONS NOW AVAILABE WORLDWIDE!

Visit http://drupalwatchdog.com/subscribe

LibreOffice Online and ownCloud
Pulling Together

Cooperation between the private company Collabora and the open source ownCloud project takes LibreOffice into the cloud. We tested the developer version and reveal what the online office solution holds for users. *By Sabine Drasnin and Heike Jurzik*

Teamwork in enterprises and community projects no longer means sitting around in meeting rooms or listening in on lengthy telephone conferences. Modern cloud solutions allow decentralized work – whether in the office, home office, or the café on the beach. Most in demand are standard office applications such as word processors, spreadsheets, and presentation tools. Ideally, the programs should look the same in the web browser as on the desktop and offer simultaneous document editing and version control in addition to the basic functions.

Microsoft Office Online [1] and Google Drive [2] have established their position in the market in recent years. The big players are now in for some competition from Collabora [3]. The Cambridge, UK, company integrates LibreOffice Online into its ownCloud Server [4], putting the free office suite on a solid file-hosting basis. In December 2015 [5], the partners announced their cooperation and shortly thereafter released a first version of the developer platform and device-independent software. The Collabora Online Development Edition (CODE) is aimed at developers and advanced users who are interesting in testing.

To Work

The developer version is available for free as a virtual machine [6], but without support by the manufacturer. Users need VirtualBox [7] to run CODE. As the template for the new VM, you choose *openSUSE (64-bit)* and then mount the VMDK file as a virtual hard drive. In the network settings, select the *Bridged Adapter*. After that, the system is ready to go. After booting, the system reveals the URL under which own-Cloud, and thus the Collabora cloud suite, is accessible.

The test team accessed different clients via the web interface. We used the latest Firefox and Chrome versions on Debian 8.4 and Ubuntu 16.04 (both 64-bit) under Windows 10 (32- and 64-bit) and Windows 7 (64-bit), as well as OS X 10.11.4. The mobile devices we used were an iPhone 6 Plus with iOS 9.3 (Safari browser) and a Tolino Shine 8.9 with Android 4.2.2 (default Android browser

and Google Chrome). The Collabora Cloud Suite allows basic editing in the three office modules – Writer, Calc, and Impress – in this early version. Upcoming versions will support advanced editing functions and collaborative work on documents.

In addition to the open document file types, ODT, ODP, and ODS, LibreOffice Online also supports the Microsoft Office formats DOC(X), PPT(X), XLS(X). Support for Apple Pages and Numbers, as well as RTF is already listed in the features, but we failed to open these documents in our lab.

The system administrator logs in to the web interface with username *admin* and password *admin*. The menu at top right provides access to personal settings and to the administration interface for managing users and groups (Figure 1). Unprivileged users also have access to the menu at top right, where they maintain their personal settings, access the online help, and log out.

The menu at top left provides entries for the list view of all files and folders (*Files*), your own or all activities, as well
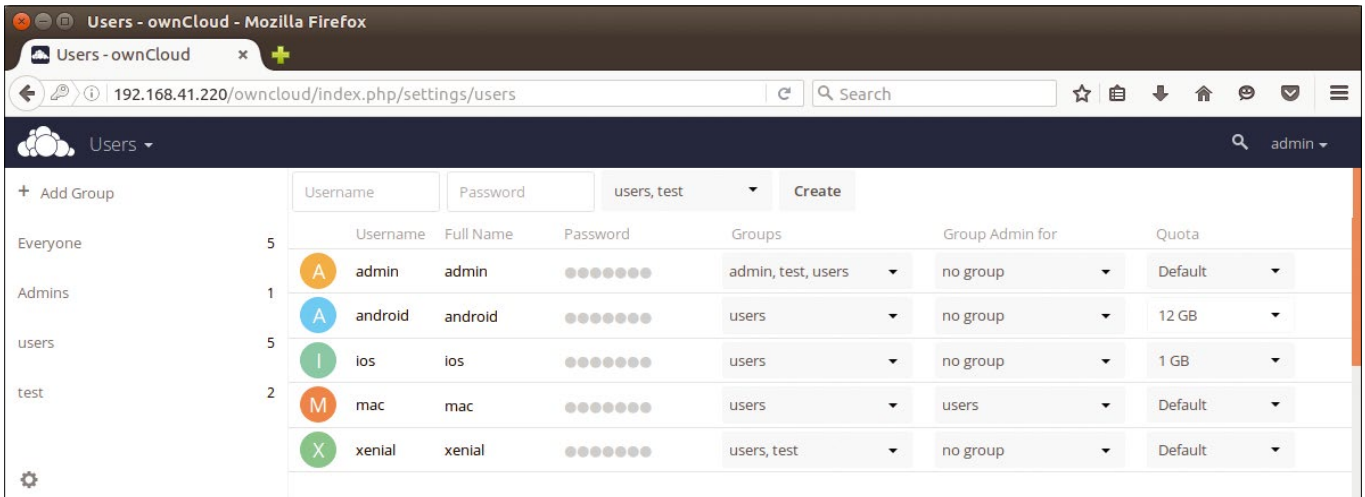
Figure 1: In user management, the admin adds new users and groups, group administrators, and a storage quota for each user.

as the gallery. The *Collabora Online* item takes you to the virtual office. The developer version includes some example documents. Buttons on the left side create new Writer, Calc, and Impress documents, and the *Upload* button sends files from the hard drive to the cloud.

In the icon view (Figure 2) it is impossible to rename or delete objects, although you can access these features in the list view by clicking on the three dots to unfold the details, where you can assign new names, download objects, or move objects to the trash (*Deleted files* at bottom left). Functions for creating copies or shortcuts are missing.

Writer

We succeeded in creating new documents in the word processor in the test and also managed to edit existing ones.

Collabora Cloud Suite did a good job of importing DOC(X) files; the software even handled headers and footers and documents with images with no problems. To insert text, you click in the document. A double click marks a word, and triple clicking selects the row. Alternatively, you can select sections of text by dragging the mouse. The text selected in this way has two slide controls, which mark the start and end of the selected range (Figure 3). Using these handles, you can decide exactly where to put the markers.

Writer offers some basic formatting settings as icons in the toolbar: bold, italic, underline, strikethrough, font color, and highlights. Additionally, you can define the font type and size, text alignment, and bulleted and enumerated lists. Styles are only available in the

drop-down menu if you created the document in the web interface or in the desktop version of LibreOffice. You cannot retroactively assign any other styles to imported DOC(X) files.

Shortcuts

Clicking on the icon with the question mark at top right opens a dialog that lists all shortcuts. The general keyboard shortcuts (e.g., Ctrl+Z for undo, Ctrl+Y for restore, and Ctrl+X for cut) work as in the desktop version.

Also, the format shortcuts (e.g., Ctrl+B for bold, Ctrl+I for italics, Ctrl+U for underline, Ctrl+1 for heading 1, Ctrl+2 for heading 2, etc.) have been taken from the desktop version.

Not all shortcuts always worked on every platform – some operating systems and work environments used shortcuts
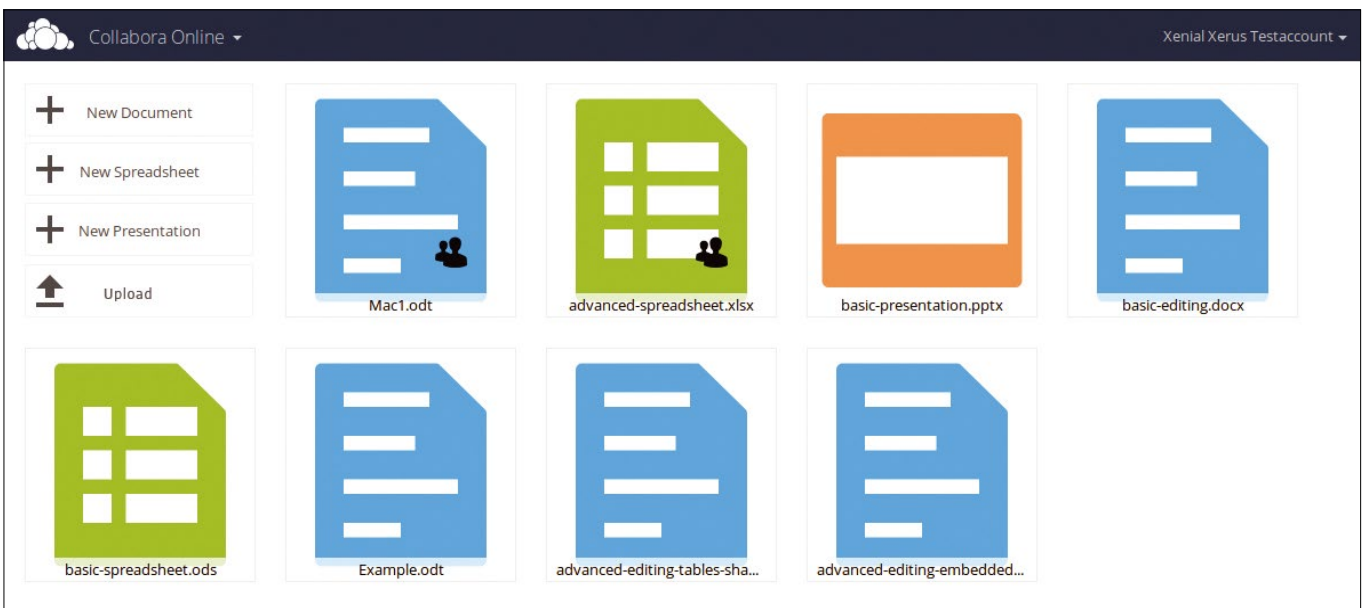
Figure 2: *Collabora Online* reveals existing documents and buttons for creating new documents.
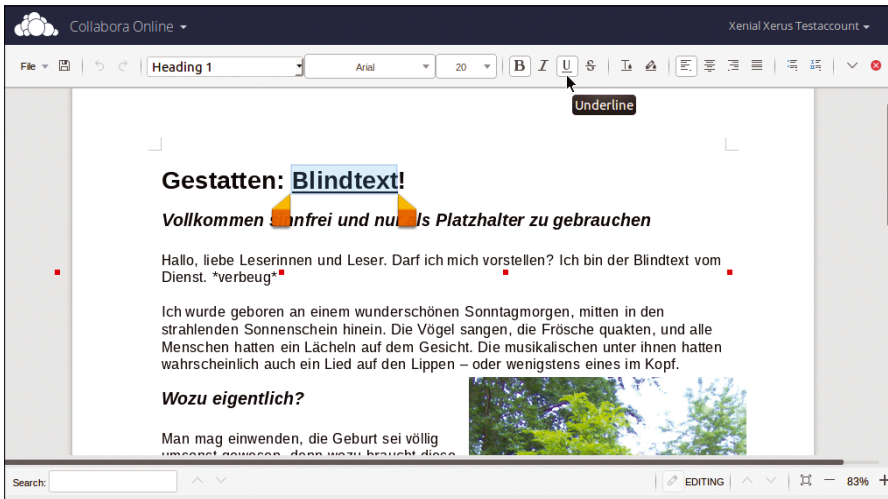
Figure 3: The Writer word processor currently only offers basic formatting options, just like the other modules. Selected text is clearly visible.

for their own purposes and intercepted the keyboard shortcuts of the online suite. This applied equally to working with the mouse: No harm in a little patience. Some format changes and input only appeared after a noticeable delay on various test computers.

Using the icon next to the question mark, you can insert graphics from your local disk into your Writer files. Unfortunately, it is not possible to use images from the ownCloud Gallery, so they are currently left hanging around in the interface. In the online word processor, you click on a graphic and drag the handles to define the size; you have no other options, such as picture frames or changing the text flow, in the current version.

LibreOffice Online stores the changes while you are still working on the document, so clicking on the icon to close the file has no unpleasant consequences. If you want to save manually, click on the floppy disk icon on the left side of the toolbar; however, you cannot store a modified document under a different name. The *File* menu, where you would suspect a suitable entry, only supports downloading the Writer file in the PDF, ODT, and DOC(X) formats.

Impress
The online version of the presentation module can handle most formatting, animations, and transitions, and in our lab displayed ODP and PPT(X) files without any trouble. Only the sound and video files caused difficulty.

The same thing applies for presentations as for text documents: Documents

you did not previously format cleanly on your computer, won't look good in the Collabora Cloud Suite.

The order of the animations needs to be correct, which means you need to define the backgrounds on master slides correctly. If the slide master uses a different background for the title slide than for the remaining pages, then backgrounds might overlap – an ugly effect that occasionally occurred in our lab.

At the bottom is a control bar that inserts, duplicates, and removes slides (Figure 4), and the icon on the left starts the presentation in full-screen mode. Impress did not display all the transitions cleanly in the uploaded files, but this is something you might be able to live with. Overly elaborate animations will just confuse viewers and the default transitions work well. More effects are

planned. If you are interested in comparing the list with Microsoft Office Online and Google Drive, see the online comparison [8].

Formatting options are no different from those for Writer: The online Impress only offers basic functionality. Users can again insert images via the icon in the toolbar. Changing the size of graphics happens, as previously, by dragging the handles.

Many options are also accessible via keyboard shortcuts, which are revealed by clicking on the question mark icon. The typical Impress commands for selecting items in the slides are especially well implemented. Changing and adding text also worked well in the test. Keep calm, though, because not everything runs smoothly.

Calc
The online version of Calc has no problems with tables prepared on the desktop or with the examples included by the developers. All charts and formatting in the imported ODS and XLS(X) files looked just as they did on the desktop computer. The spreadsheet also transferred formulas without grumbling.

However, if you want to insert formulas later on, you cannot rely on the wizard to help; even simple sums have to be entered manually. In the online suite itself, formatting options and numerous shortcuts have restrictions, which the online help explains.

The spreadsheet left us with a slightly worse impression than the word processor and presentation tool. Calc accepted
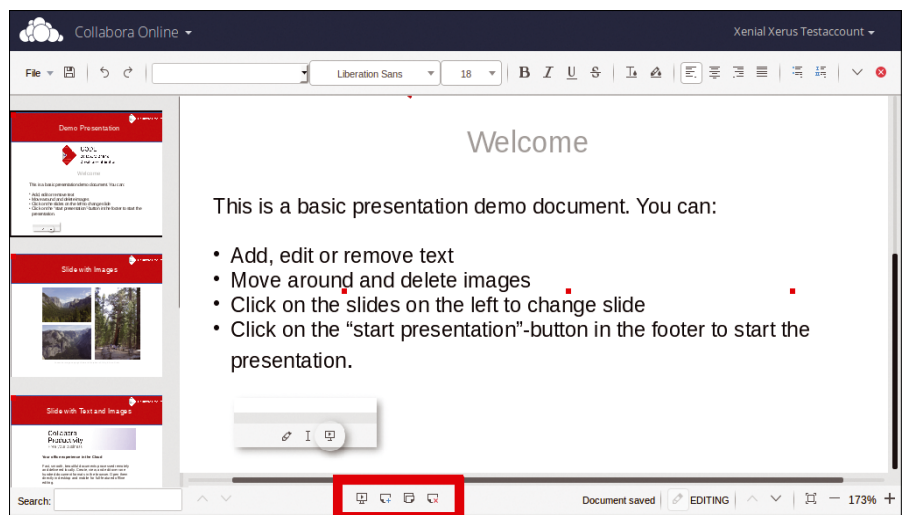
Figure 4: The online edition of Impress is quite impressive, pardon the pun. The little icons at the bottom give you access to the most important control functions.
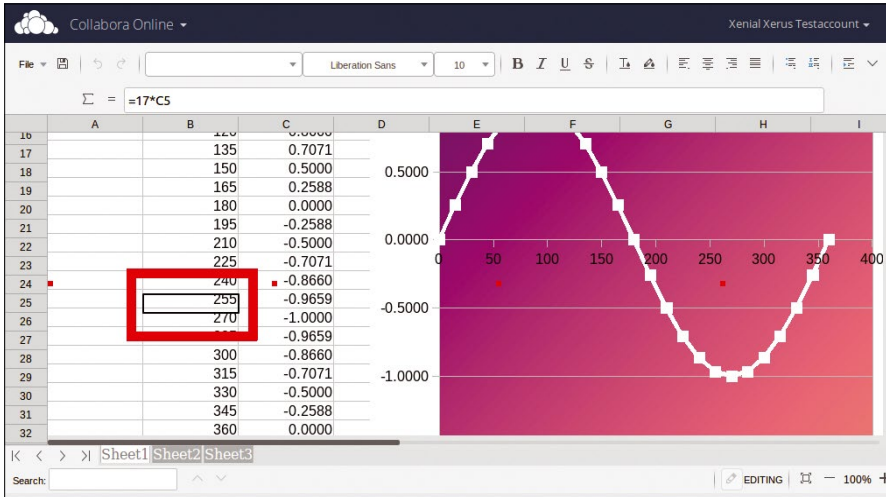
Figure 5: A crazy cursor and manual entry by hand – this could be better. So far, it's still not a convincing spreadsheet, but it's on the right track.

input on the test computers far too slowly or not at all. Also, the Undo/Redo function refused to work from time to time – the two arrows were grayed out – and the Ctrl+Z and Ctrl+Y shortcuts showed no effect.

The slightly offset cursor that occasionally appeared in Calc (Figure 5) is also ugly. In these cases, the cell itself was not selected, but the frame slid down a few pixels. The phenomenon appeared on all Windows systems and in OS X and Linux in different browsers.

Teamwork

Collaborative editing in real time is not currently possible; the developers have announced the feature for the second
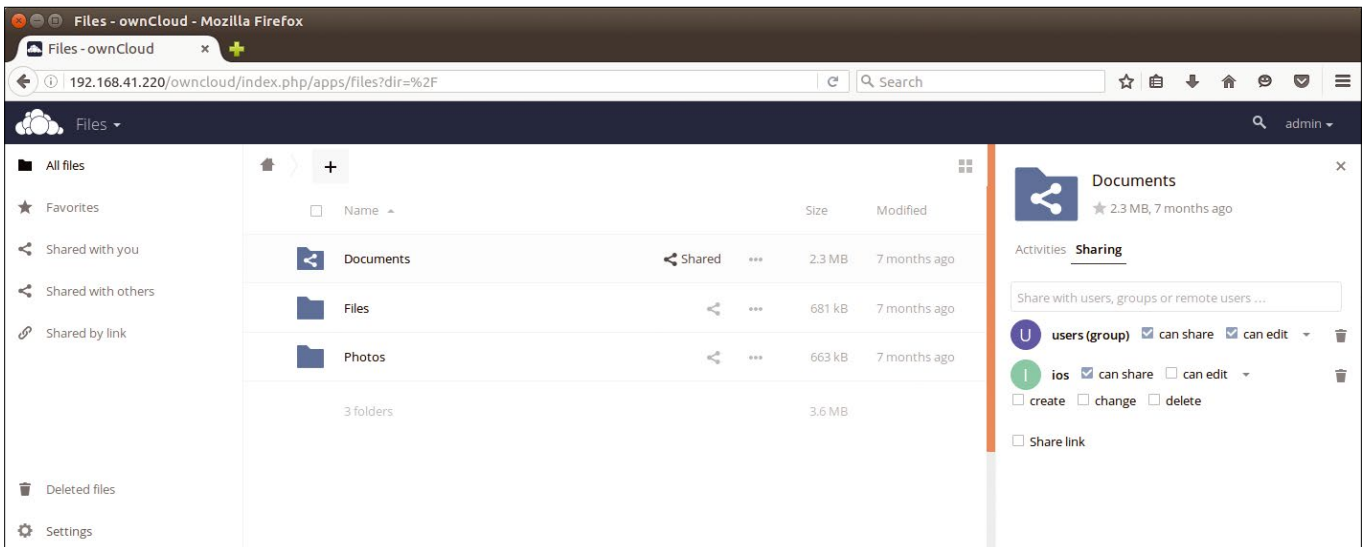
Figure 6: Users are allowed to share documents and folders with other users or groups. The owner determines whether others can only read or also edit.
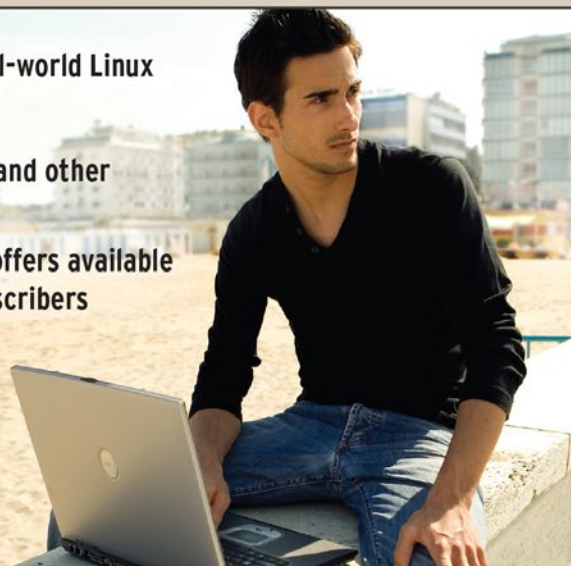
half of the year. However, you can still work together in the current development version.

To do so, you need to share folders and files to other users or groups. In the list view, select the object you want to share and then click the *Share* icon. In the box below, enter the name of the user or group; ownCloud automatically completes the information.

Using the checkboxes, owners define whether others are also allowed to edit the document. If you unfold the small arrow, you can decide in a targeted way whether to allow others to create, modify, or delete (Figure 6). The trash can icon lets your revoke the share. An own-Cloud account is required for this kind of sharing.

The *Share link* option is different. The Collabora Cloud Suite creates a temporary URL that you can share with others. You can optionally set a password and an expiration date for the public link and allow editing.

Additionally, you can open documents shared with others at the same time. Whoever opens first, sees an *EDITING* tag in the status bar at the bottom and can edit the file. The other users see a

VIEWING tag in the bar – they can only keep track of the changes made by the first user. Everything happens so fast that it looks like real time. If a second user is authorized to edit the file, they can take over by clicking on the small icon with the stylus. This user then sees an *EDITING* tag and all other users see *VIEWING* tags.

Mobile Devices

The developer version does not look good on mobile devices. The iPhone showed the ownCloud web interface in Safari and allowed access to folders, files, and shares. It was possible to edit them and to provide links to others (Figure 7). Access to Activities and the Gallery worked well.

Although we were wowed by the responsive design up to this point, the positive impression faded after selecting *Collabora Online*. The icons for each of the files were much too large, did not scale, and did not look good even in landscape mode. Creating new documents and editing existing ones also failed to work reliably. Although we succeeded at times, the rotating loading circle displayed permanently at other times.

The results on the Android tablet were slightly better. The built-in browser had no chance whatsoever, but the Chrome app at least opened the ownCloud web

interface, as well as Writer, Impress, and Calc documents. Editing was slow and prone to error, however (Figure 8).

What Remains

Although ownCloud introduced versioning in version 4, it was disabled in CODE. Precisely this feature, however, should be LibreOffice Online's unique selling point compared with others. If the developers unlock this function in the stable version, it will be the killer feature – and it even keeps an eye on quotas. By default, ownCloud never uses more than 50 percent of the free space for older file versions.

All told, the developer version left us with a positive impression. Having two proven programs work hand-in-hand is a good idea. Thanks to ownCloud, Libre-Office Online has stable underpinnings and a framework that reliably handles administrative tasks. When this issue reaches the newsstands, the first stable release of Collabora Cloud Suite should be just about released and will hopefully fix the bugs discovered in our tests. In any case, the features listed on the roadmap [9] make you want more. ∎∎∎

Figure 7: The ownCloud web interface deserves full credit for a good, responsive design. Writer, Calc, and Impress, however, were reticent.

Figure 8: Document editing only worked in the Chrome browser on Android. Unfortunately, it did not always work without error.

■ INFO

[1] Microsoft Office Online: *https://www.office.com*

[2] Google Drive: *https://www.google.com/drive*

[3] Collabora: *https://www.collabora.com*

[4] ownCloud: *https://owncloud.org*

[5] Collabora Cloud Suite announcement: *https://www.collaboraoffice. com/press-releases/collabora-and-owncloud-announce-partnership-and-release-code-for-libreoffice-online-developers*

[6] CODE developer version download: *https://www.collaboraoffice.com/ code*

[7] VirtualBox: *https://www.virtualbox.org*

[8] Transitions and animations compared with other online suites: *https://www.collaboraoffice.com/ wp-content/uploads/2016/01/Online-Office-Suites-presentation-transitions-and-animations-comparison-chart.pdf*

[9] Collabora Cloud Suite overview: *https://www.collaboraoffice.com/ solutions/collabora-cloudsuite*

Build your own crawlers

Spider, Spider
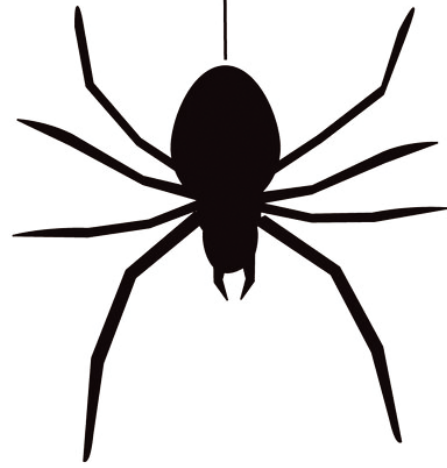
Scrapy is an open source framework written in Python that lets you build your own crawlers with minimal effort for professional results. *By Andreas Möller*

A crawler demonstrates the capabilities of version 1.0 of the Scrapy framework [1] running under Python 2.7 [2]. Scrapy is an open source framework for extracting data from websites. It recursively crawls through HTML documents and follows all the links it finds.

In the spirit of HTML5, the test created in this article is designed to reveal non-semantic markup on websites. The crawler counts the number of words used per page, as well as the number of characteristic tag groups (Table 1), saving the results along with the URL in a database.

To install the required packages, I used the Debian 8 Apt package manager:

```
apt-get install python-pip ⤷
        libxslt1-dev python-dev ⤷
        python-lxml
```

The packages include the Python package manager (Pip), the libxslt library along with the header files, the Python header files, and the Python bindings for libxml and libxslt. Because Debian 8 comes with Python 2.7 and libxml pre-installed, you can install Scrapy as follows:

```
pip install scrapy
```

Unlike Apt, Pip installs as the latest Scrapy version for Python 2.7 from the Python package index [3].

Test Run
To begin, open an interactive session in the Scrapy shell by entering `scrapy shell`

(Figure 1). Next, send a command to the Scrapy engine to tell the on-board downloader to read the German *Linux-Magazin* homepage through an HTTP request and transfer the results to the `response` object (Figure 2):

```
fetch('http://www.linux-magazin.de')
```

Figure 3 demonstrates in detail how the components of the Scrapy architecture work together. This illustration makes it clear that the engine does not talk di-

TABLE 1: Definition of Stored Metrics

Metric	Meaning
keywords	Number of words in the *<title>* tag
words	Number of all words except keywords
relevancy	Frequency of keywords in the total number of words
tags	Total number of all tags
semantics	Total number of all semantic tags
links	Total number of all *<a>* tags with an *<href>* attribute
injections	Number of third-party resources

```
redaktion@debian: ~
                        redaktion@debian: ~ 79x26
redaktion@debian:~$ scrapy shell
2016-05-06 16:38:16 [scrapy] INFO: Scrapy 1.0.6 started (bot: scrapybot)
2016-05-06 16:38:16 [scrapy] INFO: Optional features available: ssl, http11
2016-05-06 16:38:16 [scrapy] INFO: Overridden settings: {'LOGSTATS_INTERVAL': 0
, 'DUPEFILTER_CLASS': 'scrapy.dupefilters.BaseDupeFilter'}
2016-05-06 16:38:16 [scrapy] INFO: Enabled extensions: CloseSpider, TelnetConso
le, CoreStats, SpiderState
2016-05-06 16:38:16 [scrapy] INFO: Enabled downloader middlewares: HttpAuthMidd
leware, DownloadTimeoutMiddleware, UserAgentMiddleware, RetryMiddleware, Defaul
tHeadersMiddleware, MetaRefreshMiddleware, HttpCompressionMiddleware, RedirectM
iddleware, CookiesMiddleware, ChunkedTransferMiddleware, DownloaderStats
2016-05-06 16:38:16 [scrapy] INFO: Enabled spider middlewares: HttpErrorMiddlew
are, OffsiteMiddleware, RefererMiddleware, UrlLengthMiddleware, DepthMiddleware
2016-05-06 16:38:16 [scrapy] INFO: Enabled item pipelines:
2016-05-06 16:38:16 [scrapy] DEBUG: Telnet console listening on 127.0.0.1:6023
[s] Available Scrapy objects:
[s]   crawler    <scrapy.crawler.Crawler object at 0x7f0a4abc5b50>
[s]   item       {}
[s]   settings   <scrapy.settings.Settings object at 0x7f0a4abc5ad0>
[s] Useful shortcuts:
[s]   shelp()          Shell help (print this help)
[s]   fetch(req_or_url) Fetch request (or URL) and update local objects
[s]   view(response)    View response in a browser
>>>
```

Figure 1: In the Scrapy shell, you can test commands interactively.

Lead Image © mtkang, 123RF.com

rectly to the downloaders but first passes the HTTP request to the scheduler (Figure 3, top). The downloader middleware (Figure 3, center right) modifies the HTTP request before deployment. `Cook-iesMiddleware`, which is enabled by default, stores the cookies from the queried domain, whereas `RobotsTxtMiddleware` suppresses the retrieval of documents blocked for crawlers by the `robots.txt` [5] file on the web server.

Tracker

Scrapy evaluates the document components by interactively querying and investigating them via the response object, as shown in Figure 2. The selection is made either with the help of CSS selectors [6], as in jQuery, or with XPath expressions [7], as in XSLT. For example, first enter the command,

```
response.xpath(⤶
  '//title/text()').extract()
```

as shown in Figure 2 to call the `xpath()` method. The `//title` subexpression first selects all *< title >* tags from the HTML document, and `/text()` selects the text nodes that follow. The `extract()` method transfers the results set to the Python list:

```
[u'Home \xbb Linux-Magazin']
```

Using the expression

```
len(response.xpath(⤶
  '//a/@href').extract())
```

you can extract the values of the *< href >* attributes of all *< a >* tags in a list. Their length is discovered by the Python `len()` function; in this case, there are *215* (see Figure 2).

Getting Started

A sample application can be compiled with a little knowledge of Scrapy. The command

```
scrapy startproject mirror
```

lays the foundations by creating a matching directory structure (Listing 1). The application reaches the user after changing to the `mirror` project directory. Empty files with the name `__init__.py` are purely technical in nature [8]. The

`spider` and `pipeline` classes can be found in subdirectories of the same name. Scrapy stores the results in the `results` directory and the associated reports in the `reports` directory.

A few listings will be added to the skeleton project later. The `mirror/utils.py` file from the last line of Listing 1 stores the helper functions. Listing 2 shows the contents of this file.

The global settings for the project are also in Python format and belong in `mirror/settings.py` (Listing 3). Scrapy thus creates the variables in the first three lines, their capitalization is reminiscent

of constants in C, although they are not available in Python.

Line 1 stores the name Scrapy sends to the requested web server instead of the browser identifier in the header of the HTTP requests. Lines 2 and 3 are inherent in the Scrapy system and require no change. The variables that follow store application-specific constants.

Transport Workers

Before sending the requested data into the item pipeline (Figure 3, left), Scrapy converts it to an item object. This modification helps to format and validate the

Figure 2: The on-board downloader bundles the Linux-Magazin site into a response object.

Figure 3: The Scrapy engine delegates tasks to different components, like the spider, the item pipelines, and middleware [4]. Twisted, an event-driven network framework, works in the background.

LISTING 1: mirror Sample Project

```
|- scrapy.cfg
|- mirror:
  |- __init__.py
  |- items.py
  |- pipelines
    |- __init__.py
    |- filter.py
    |- normalize.py
    |- store.py
  |- reports
    |- attr.py
  |- results
    |- 2016032210001458637243.sqlite3
  |- settings.py
  |- spiders
    |- __init__.py
    |- attr.py
  |- utils.py
```

LISTING 2: mirror/utils.py

```
01 from urlparse import urlparse
02
03 def optvalue(alist, key, default=[]):
04   if key in alist:
05     return alist[key]
06   return default
07
08 def domain(url):
09   return urlparse(url).netloc
10
11 def join(tags):
12   return "|".join(['//'+tag for tag in tags])
13
14 def relevance(kds, wds):
15   if len(kds) == 0 or len(wds) == 0:
16     return 0
17   return reduce(lambda acc, kw: float(wds.count(kw)) + acc, kds, 0)/len(kds + wds)
18
19 def is_absurl(url):
20   return reduce(lambda acc, p: url.startswith(p) or acc, [u'http://',
                 u'https://', u'//'], False)
```

LISTING 3: mirror/settings.py

```
01 BOT_NAME = 'mirror'
02 SPIDER_MODULES = ['mirror.spiders']
03 NEWSPIDER_MODULE = 'mirror.spiders'
04 ITEM_PIPELINES = {
05     'mirror.pipelines.normalize.Words': 300,
06     'mirror.pipelines.filter.Injections': 400,
07     'mirror.pipelines.store.Attributes': 500
08 }
09 RESULTS = 'mirror/results/'
10 MEDIA_TAGS = ['video', 'audio', 'img', 'canvas']
11 INJECT_TAGS = ['script/@src', 'img/@src', 'video/@src', 'audio/@src', 'iframe/@src', 'embed/@src', 'link/@href']
12 SEMANTIC_TAGS = ['html', 'head', 'title', 'meta', 'link', 'body', 'header', 'footer', 'nav', 'article', 'aside',
                'section', 'h1', 'h2', 'h3', 'h4', 'p', 'a', 'ul', 'ol', 'li', 'dl', 'dt', 'figure', 'table', 'th',
                'tr', 'td', 'video', 'audio', 'form', 'input',  'label', 'button']
```

application data. The item class of the sample application is `mirror/items.py` (Listing 4).

Line 1 explicitly imports the required base classes `Item` and `Field` from the Scrapy package. Line 3 declares the `Item` class Attributes, which points to the parent class `Item` in the parentheses. The final colon introduces a block to which all following lines with the same indentation belong.

Scrapy does not define the application data directly in the attributes but stores them in objects of type `Field` (lines 4 to 11). The constructor of `Field` comprises, as usual, the class name and pair of round brackets.

Thanks to the call parameter, the `Field` object first runs the data to be saved through the `__call__()` method, to which the passed object of type `Split` belongs. The class instantiates the code in lines 13-15. The signature of `__call__`() is predefined, as is the case for all methods that Scrapy automatically invokes. The method takes a list of character strings and breaks the more complex list expression down into individual words in line 15.

The first `for` loop iterates over all `value` items in the `values` list. The second loop uses `split()` to tackle the divided `value` string. With each pass, the script uses the `word` variable (to the right of the opening square bracket) to add a word to the resulting list.

Industrious Spider

Within the Scrapy application, the aforementioned spiders manage the show (Figure 3, center). They save part of the application code, which, as shown in Figure 2, calls the `fetch()` function in the interactive session and evaluates the `response` objects. Listing 5 shows the spider waiting in the `mirror/spiders/attr.py` application. In addition to `CrawlSpider` shown here, Scrapy offers the generic spiders `XMLFeedSpider` and `CSVFeedSpider`.

Lines 1-5 import the needed classes and functions. The rest of the listing defines the spider class, which is derived from `CrawlSpider`. The crawler launched with `scrapy crawl attr` selects, instantiates, and invokes the spider class by name (i.e., the value of the `name` attribute in line 8). Starting from line 9, the object marches recursively through the documents. The `rules` attribute typically

stores rules, much like a router. It then assigns the URLs to different callback functions. The assignment sets the default route in the first element of the list by passing in a `LinkExtractor()` object without a path to the `Rule` object. Then `callback` defines `parse_page()` as the callback function (line 13).

Before continuing, objects of `CrawlSpider` type first read all the page addresses one by one from the list in the `start_urls` attribute. They hand over the response objects to the callback functions. The `allowed_domains` statement in line 11 is optional. It restricts queries to the listed domains.

Immediately in line 13 `parse_page()` picks up the `response`object as the second argument and puts it in the `resp` variable. The next line instantiates the `ldr` variable, a container object of `ItemLoader` type that can hold other objects. It first takes care of initializing the `item` object of type `Attributes` handed over in Listing 4.

Copy Shop

Lines 15-22 of Listing 5 copy the values from the `response` object to the attributes of the `item` object. Line 15 uses the `add_value()` method to assign the URL of the `response` object to an attribute with the same name in the `item` object. Lines 16 to 22 use `add_xpath()` to copy document components to the listed attributes based on the XPath expressions passed in.

Line 16 uses `//title` to extract all *title* > tags, and the `string()` XPath function retrieves the respective text values. The `add_` functions manage their results as lists. Line 17 retrieves all the words from the HTML document in a similar way. The

`[name()!='title']` expression only chooses the tags not called < *title* > from the selection of all tags (`//*`).

As shown in Figure 2, lines 18 and 19 copy all links and tags to the `item` object. The next three lines evaluate the `MEDIA_TAGS`, `SEMANTIC_TAGS`, and `INJECT_TAGS` lists from Listing 3. The `join()` helper function from `mirror/utils.py` transforms these into XPath expressions. To be more precise, `join()` formats the given list of tags `['a', 'b']` as `//a | //b`. The pipe symbol represents the OR operator in XPath. The spider uses the `get()` method of the `settings` objects to find the three lists in an attribute of the same name.

Line 23 ends the initialization of the `item` object, by calling the `load_item()` method. Empty attributes would cause an error. The `return` statement passes the `item` object into the item pipeline.

Pneumatic Tube

The item pipeline continues to process the `Item` objects; the `ITEM_PIPELINES` variable configures the pipeline in lines 4-8 of the sample application in Listing 3. It passes each of the `Item` objects one by one to an object of pipeline classes `Words` (300), `Injections` (400), and `Attributes` (500), then modifies or stores them and pushes them forward.

Listing 6 shows the code for the `Words` class, which checks and standardizes the words for the following evaluation. Scrapy creates and binds the pipeline objects by calling the `process_item()` method for each `item` object (line 2). The method expects the `item` object as the second argument and the `spider` object, a reference to the calling spider, as the third object.

The next two lines overwrite the values of the `keywords` and `words` attributes. In line 4, the `filter()` function fishes all the items from the `item[key]` list, for which the lambda function (`lambda wd: wd.isalnum()`) returns a true value. This operation affects strings of purely alphanumeric characters.

The second lambda function – `map()` – then formats the words in the resulting set to lowercase. The `return` statement in the last line hands over the `item` object to the next pipeline object (Listing 7), which considers the possible effect of foreign content on the current browser session.

LISTING 4: mirror/items.py

```
01 from scrapy import Item, Field
02
03 class Attributes(Item):
04   url  = Field()
05   keywords = Field(=Split())
06   words = Field(=Split())
07   medias = Field()
08   links = Field()
09   tags = Field()
10   semantics = Field()
11   injections = Field()
12
13 class Split():
14   def __call__(self, values):
15     return [word for value in values for word
                  in value.split()]
```

LISTING 5: mirror/spiders/attr.py

```
01 from scrapy.spiders import CrawlSpider, Rule
02 from scrapy.linkextractors import LinkExtractor
03 from scrapy.loader import ItemLoader
04 from mirror.items import Attributes
05 from mirror.utils import join
06
07 class Attr(CrawlSpider):
08   name = "attr"
09   rules = [Rule(LinkExtractor(), callback='parse_page')]
10   start_urls = ["http://localhost"]
11   allowed_domains = ["localhost"]
12
13   def parse_page(self, resp):
14     ldr = ItemLoader(item=Attributes(), response=resp)
15     ldr.add_value('url', resp.url)
16     ldr.add_xpath('keywords', 'string(//title)')
17     ldr.add_xpath('words', "string(//*[name()!='title'])")
18     ldr.add_xpath('links', '//a/@href')
19     ldr.add_xpath('tags', '//*')
20     ldr.add_xpath('medias', join(self.settings.get('MEDIA_TAGS')))
21     ldr.add_xpath('semantics', join(self.settings.get('SEMANTIC_TAGS')))
22     ldr.add_xpath('injections', join(self.settings.get('INJECT_TAGS')))
23     return ldr.load_item()
```

The `Injections` pipeline class from Listing 7 relies on the helper functions `is_absurl()` and `domain()`, which come from `mirror/utils.py`, to reduce the list of resources bound to foreign content.

To do this, the `process_item()` method overwrites the attribute using the filtered list, which the list expression in line 5 creates. The first `for` loop only reads the URL of the page; the second `for` iterates over the tags to be injected. If the `if` statement determines that the attribute is an absolute URL different from the current domain, the loop picks up the resource from the `other` variable. The `return` statement hands over the item altered in this way to the last link in the pipeline (Listing 8), which reduces the results for later evaluation and stores them in a database.

The `Attributes` pipeline class (Listing 8, line 6) evaluates the item object and stores the results in a SQLite database file [9]. The free SQL-compatible database framework does not require server processes and supports all common programming languages.

Listing 8 writes all data directly and synchronously into the database file. Line 1 binds the correct driver for Python, and the next three lines import the required functions from the standard packages `os.path`, `time`, and `mirror/utils.py`.

As its second parameter, the `__init__()` constructor accepts the `path` with which Python opens the SQLite database. Scrapy uses the `from_crawler()` class method (lines 10-12) to instantiate an object. A look at the method shows that `from_crawler()` adds the `crawler` object to its parameter list as a reference to the settings from Listing 3. First, it reads the value of the `RESULTS` variable (Listing 3, line 9); then, it adds that value to the constructor call in lines 7 and 8 (Listing 8). Finally, `gmtime()` and `strftime()`, in combination with a string formatting variant, generate a timestamp for the file name.

LISTING 6: mirror/pipelines/normalize.py

```
01  class Words(object):
02    def process_item(self, item, spider):
03      for key in ['words', 'keywords']:
04        item[key] = map(lambda wd: wd.lower(), filter(lambda
                        wd: wd.isalnum(), item[key]))
05      return item
```

LISTING 7: mirror/pipelines/filter.py

```
01  from mirror.utils import is_absurl, domain
02
03  class Injections(object):
04    def process_item(self, item, spider):
05      item['injections'] = [other
06        for own in item['url']
07        for other in item['injections']
08        if domain(other) != domain(own) and is_absurl(other)
09      ]
10      return item
```

LISTING 8: mirror/pipelines/store.py

```
01  import sqlite3
02  from os.path import join
03  from time import gmtime, strftime
04  from mirror.utils import relevance, optvalue
05
06  class Attributes(object):
07    def __init__(self, path):
08      self.db = path
09
10    @classmethod
11    def from_crawler(cls, crawler):
12      return cls(join(crawler.settings.get('RESULTS'), "%s.sqlite3" %(strftime('%Y%m%d%H%M%s', gmtime()))))
13
14    def open_spider(self, spider):
15      self.conn = sqlite3.connect(self.db, isolation_level = None)
16      self.cur = self.conn.cursor()
17      self.cur.execute("CREATE TABLE Attributes (url text PRIMARY KEY, keywords int, words int, relevancy int, tags int,
                    semantics int, medias int, links int, injections int)")
18
19    def close_spider(self, spider):
20      self.conn.close()
21
22    def process_item(self, item, spider):
23      self.cur.execute('INSERT INTO Attributes VALUES (?,?,?,?,?,?,?,?,?  )', (item['url'][0], len(item['keywords']),
                    len(optvalue(item, 'words')), relevance(optvalue(item, 'keywords'), optvalue(item, 'words')),
                    len(optvalue(item, 'tags')), len(optvalue(item, 'semantics')), len(optvalue(item, 'medias')),
                    len(optvalue(item, 'links')), len(optvalue(item, 'injections'))))
24      return item
```

TABLE 2: Interpretation of Acquired Data

Size	Computation	Interpretation
Relevancy	–	Measure of the credibility of the title
Entropy	(*words*+*semantics*)/ (*words*+*tags*)	Non-semantic tags such as *div* or *span* reduce the information content
Expressivity	*semantics*/*tags*	Semantic tags improve the functional classification of document components
Richness	*medias*	Media enrich the content
Reliability	*links*/*words*	Links vouch for the credibility
Mutability	*injections*	External resources alienate the page

The `open_spider()` method (lines 14-17) calls the Scrapy engine once only, in the style of a callback function, when creating the spiders. It creates and stores a database connection in the `conn` attribute in line 15. Specifying `isolation_level = None` tells the driver to create each SQL statement at once persistently in the database file. Line 16 creates and stores the database `cursor` object that runs database operations. The operations also include the SQL command that creates the result table:

```
CREATE TABLE Attributes (
  url text PRIMARY  KEY, keywords int,
  words int, relevancy int, tags int,
  semantics int, medias int,
  links int, injections int)
```

The `process_item()` method from line 22 combines the values of the `item` object with the `Attributes` table according to Table 2 using the SQL `INSERT` command. The question marks are replaced by the values of the next tuple in the parentheses. The `len()` function counts the length of the parsed lists several times. The helper function `optvalue()` swaps `None` values for empty lists; `relevance()` determines the incidence of all `keywords` in the remaining text of the website.

Evaluation

As discussed earlier in the article, you launch the crawler at the command line from within the `mirror` project directory:

```
scrapy crawl attr
```

Listing 9 shows the SQL query that generates the report shown in Figure 4 according to Tables 1 and 2. The strength of SQL is revealed in the compact style of expression, which is very similar to sentences. However, converting the types and formatting requires tedious typing.

The endogenous page factors evaluate a web page from various perspectives. The derived entropy value is a measure of the average information content of the page. The values are worded identically but are not identical to the corresponding terminology from information theory [10].

In the sample application, they describe only how non-semantic tags like `div` and `span` dilute the content. If the entropy value were 1, the generic spider would achieve better results. The average of 0.837 in Figure 4 indicates some scope for improvement.

Conclusion

Programming with Scrapy is fun and offers surprising insights. Thanks to the cleverly chosen modularization and good documentation, users can focus on extracting and accumulating data. If you delve deeper into Scrapy, you will also see the multitude of aspects it covers and the professional approach the framework pursues. ∎∎∎

▌ INFO

[1] Scrapy framework: *http://scrapy.org*

[2] Python: *https://python.org*

[3] Python package index: *pypi.python.org*

[4] Scrapy docs: *http://doc.scrapy.org/ en/latest/topics/architecture.html*

[5] Robots exclusion standard: *https://en.wikipedia.org/wiki/ Robots_Exclusion_Standard*

[6] CSS selectors: *https://api.jquery. com/category/selectors/*

[7] XPath expressions: *https://www.w3.org/TR/xpath/*

[8] Meaning of `__init__.py`: *http://stackoverflow.com/questions/ 448271/what-is-init-py-for*

[9] SQLite: *http://sqlite.org*

[10] Information theory: *https://en.wikipedia.org/wiki/ Information_theory*

	Ø	0.060	0.837	0.296	1.200	0.039	1.000
url		**relevancy**	**entropy**	**expressivity**	**richness**	**reliablity**	**mutability**
http://pamoller.com/		0.041	0.858	0.603	1.000	0.133	1.000
http://pamoller.com/Caches-mit-Materialisierten-Sichten.html		0.026	0.937	0.285	2.000	0.002	1.000
http://pamoller.com/storage.html		0.022	0.858	0.163	0.000	0.011	1.000
http://pamoller.com/mboxfilter.html		0.023	0.893	0.220	0.000	0.013	1.000
http://pamoller.com/PGConf-2011.html		0.016	0.796	0.334	4.000	0.008	1.000
http://pamoller.com/Kontakt.html		0.000	0.439	0.245	0.000	0.176	1.000
http://pamoller.com/VirtualBox.html		0.056	0.926	0.392	5.000	0.008	1.000
http://pamoller.com/xmlformatter.html		0.023	0.850	0.199	0.000	0.011	1.000
http://pamoller.com/restfooly.html		0.024	0.896	0.282	0.000	0.012	1.000
http://pamoller.com/setTimeout-vs-setInterval.html		0.060	0.918	0.233	0.000	0.012	1.000

Figure 4: The page factors on the basis of the relative frequencies of the appropriate endogenous variables.

▌ LISTING 9: Report SQL Query

```
SELECT url, printf('%.3f', relevancy), printf('%.3f', cast((words + semantics) as float)/(words + tags)) as entropy,
    printf('%.3f', cast(semantics as float)/tags) as expressivity, printf('%.3f', cast(medias as float)) as richness,
    printf('%.3f', cast(links as float)/words) as reliablity, printf('%.3f', cast(injections as float)) as mutability FROM
    Attributes
```

Subscribe today and join the revolution!

Each issue of Raspberry Pi Geek is an adventure, with ingenious applications and cool projects for Raspberry Pi, Arduino, and other maker-board systems!

Discover the secrets that will empower you to envision and build your own Raspberry Pi inventions.

Features

Special articles on new technologies and topics of interest to the Rasp Pi community.

RASPBERRY PI GEEK

raspberry pi GEEK

In this issue we celebrate the Raspberry Pi's fourth birthday. We also explore some new features of the newest Pi and present some projects that take advantage of these faster, more capable machines, such as 3D modeling and distributed compiling.

Your Pi is ready!!!

8 Rasp Pi 3 in Detail: Faster with more functionality: What can you do with an RPi3?

90 Paddleboat: Program your own Android app to control a paddleboat or rover.

Features

8 Inside the Raspberry Pi 3
The Raspberry Pi 3 changes the game with its fast, 64-bit CPU and support for WiFi and Bluetooth.

20 Raspberry Pi Turns 4
The Raspberry Pi celebrates its fourth anniversary. We review the history of the nifty nanocomputer.

Oooh, so very impressive!

Service

3 Welcome
6 DVD
98 Masthead

Skills

68 SwitchDoc Labs – Solar Tracking
How much extra power can a solar panel collect by tracking the sun?

74 Scripting
Linux has a powerful set of programming tools that lets you manipulate data without the need to install additional software.

80 Man Pages
Man pages provide essential information but may seem cryptic if you're not familiar with their structure. We introduce you to the native help system for Linux commands.

ISSUE 17

RASPBERRY PI GEEK

4

---

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

$59.95 £37.50 €44.90   shop.linuxnewmedia.com

Highlights

28 **distcc:** Compile Rasp Pi programs on more powerful computers.

68 **Solar Tracking:** Sun-tracking vs. stationary solar panels.

74 **Scripting:** Massage your data with native Linux tools.

80 **Man Pages:** These handy helpers offer immediate assistance with Linux commands.

Projects

Do-it-yourself, real-world projects that let you learn by doing.

ISSUE 17

DVD

Every print issue comes with a free DVD!

raspberry pi **GEEK**

NOOBS 1.9.0
- Raspbian "jessie"
- OpenELEC RPi2
- OSMC RPi2
- RISC OS
- Windows 10 IoT Core
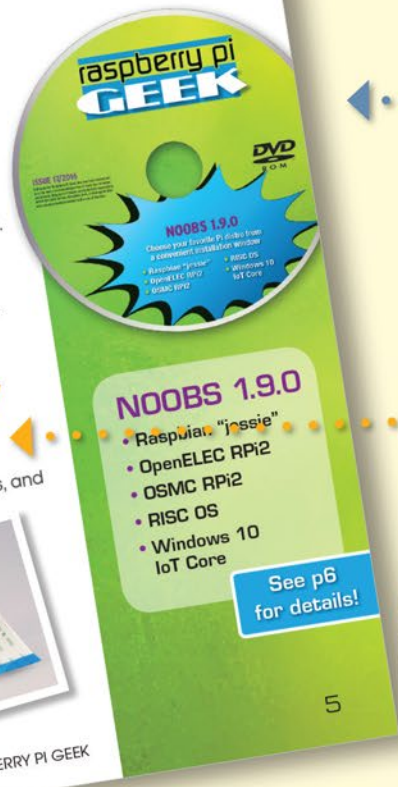
Community

Interviews and reports on Raspberry Pi meet-ups around the world.

Get your Pi on!

Kid Stop

Special projects for kids, including a new Scratch programming exercise in every issue.

Skills

Timely tutorials to help you build your skills with Linux and other underlying technologies.

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*

Mounting and Unmounting Network Storage Devices

? I'm using a small NAS storage server as network drive, which I mount the "Unix way" by NFS (Network File System), rather than using a Windows-compatible Samba client, which also would be supported by the device. That way, the mount point is accessible to all programs, not only the file manager or special clients that can use the *smb://* protocol.

The command for mounting that I use under the directory name /media/volume1 is:

```
sudo mount -t nfs 10.0.0.20:/volume1 ⤶
  /media/volume1
```

Now, when suspending and disconnecting my computer from the network for traveling, I sometimes forget to unmount the NFS directory /media/volume1, which results in all programs freezing when trying to access that directory. Trying to unmount in that situation with

```
sudo umount /media/volume1
```

does not work because the *device is busy*. Even when no processes are accessing the directory, unmounting fails because the server is not reachable.

Is there an "official" way to unmount NFS volumes when the server is not reachable?

💡 NFS was built to be reliable even in the case of network dropouts or temporary server disconnects; the client will wait for a long, sometimes even an unlimited time for the server to respond again. Even signals like TERM (15) and KILL (-9) won't terminate a process that waits for an answer from the NFS server.

Furthermore, you can even reboot the NFS server and the client still won't terminate. This is the desired behavior in NFS design.

Some mount options, like -o intr or -o timeo=n or -o retrans=n (see the NFS man page) change that behavior, but still you won't be able to unmount the NFS filesystem when the server is unreachable.

However, an easy trick allows you to work around the protocol: Just set the IP address of the former NFS server as the secondary IP address of your network card or loop back device first:

```
sudo ifconfig lo:1 10.0.0.20 ⤶
  netmask 255.255.255.255
```

For the client to detect the NFS server (it does not matter whether or not any directories are exported), start the port mapper and, if installed, the NFS kernel-based server:

```
sudo /etc/init.d/portmap start
```

or

```
sudo /etc/init.d/rpcbind start
```

then:

```
sudo /etc/init.d/nfs-kernel-server start
```

(The `/etc/exports` file should exist; otherwise, the init script may decide not to start the actual NFS server.) An `rpc.mountd` process should now be running, answering to `mount` and `umount` requests from the client.

The NFS client will now act as if the former NFS server is back online (even though there is really a new NFS server running on the client computer now), send a detach command on `umount` and flush all client accesses (provided no processes are still actively using the mount point):

```
sudo umount -l /media/volume1
```

I use the `-l` (lazy) option of `umount` here, which will allow running processes to continue accessing the directory but hide the mount point from new processes and remove the `mtab` entry, so it is not listed as a mounted volume anymore in `/proc/mounts` or in the output of `mount` or `df` with no parameters.

Another tip: To avoid too lengthy command lines for mounting NFS volumes or Samba shares, `/etc/fstab` entries can be

helpful. Note that `intr` and `nolock` options for NFS allow you to cancel a read/write command by hitting Ctrl + C and avoids contacting the NFS locking process on the server side. Also, for NFS on Linux, the `sync` option (which is known to slow down write access to local media) can actually make NFS more responsive because it avoids buffering too much data in long writes, thus avoiding a traffic jam on wireless networks (Listing 1).

With these entries, the network storage directories can be mounted just by typing

```
mount /media/volume1
```

or

```
mount /media/backup
```

without `sudo`, as normal user. ∎∎∎

LISTING 1: /etc/fstab

```
01 /etc/fstab:
02
03 # Mounting an NFS share with options (see nfs manpage)
04 # "sync" avoids latency due to excessive buffer usage.
05 10.0.0.20:/volume1 /media/volume1 nfs sync,users,noauto,rw,tcp,nolock 0 0
06
07 # Similar for SAMBA, using a login/password combination for accessing the
     "backup" share
08 # uid and gid should match those of the user on the client side for easy
     permission handling
09 # username and password are given in cleartext here, which is insecure;
     otherwise mount will ask for them
10 //10.0.0.20/backup /media/backup cifs users,noauto,username=share_
     user,password=share_passwd,uid=1000,gid=1000 0 0
```

Use your Chromebook with Linux

Unleashed

Chromebooks are firmly locked in the Google jail, but with the right know-how, you can break out of vendor lock-in and operate the devices with free software. *By Erik Bärwaldt*

Chromebooks have established themselves as a low-cost alternative to conventional entry-level laptops. Devices from many manufacturers and with many display sizes are available; however, all models are tightly tied to the Chrome OS operating system by Google.

Some potential buyers, toying with the idea of investing in such a device, are discouraged by the very limited capabilities of the operating system despite what is now quite powerful hardware. And, serious privacy concerns are another drawback to the system. Google is one of the largest collectors of data worldwide and makes things easy for itself by closely tying Chrome OS to its online services to extract personal data from users and thus generate profiles.

As a US corporate group, Google is forced to hand over the data to US authorities when ordered to do so. But the close bond between the operating system, Google's services, and the hardware is not hard-wired, so resourceful minds from the free developer community have broken these restraints and customized Linux for Chromebooks. The easiest approach is to create a dual-boot system using the ChrX tools [1]. Alternatively, you can use a separate Linux derivative in a window on Chrome OS in the form of Crouton [2].

Both of these options have serious drawbacks in that the parallel installation of two systems takes up unnecessary space on the relatively small Chromebook storage device. And, the use of Chrome OS rules out adequate data protection.

Therefore, the complete deletion of Chrome OS through an appropriately customized distribution is advisable. For cases in which users try to completely oust the system from their device, Google has taken numerous precautions to make it enormously difficult for less-skilled computer users to install alternative software.

Hurdles

Installing an alternative operating system instead of Chrome OS on a Chromebook involves not only replacing the device's firmware with a free alternative, but typically also includes some manipulation of the hardware. Most devices are equipped with a special jumper or a screw on the motherboard that switches the system software to read-only mode.

Write access is only allowed if you close the jumper or remove the screw; you can then modify the firmware. In many cases, both components are in place, and the manufacturer is likely to have glued a warranty seal over the corresponding screw so that the warranty is void in the case of removal. And, some companies do not disclose the position of the components on the motherboard of their devices, so you need to find these components by searching on the Internet.

Another problem is the lack of uniform hardware: Firmware modifications and thus replacement of the basic input/output system (BIOS) are by no means possible on all Chromebooks: Free solutions do not take into account certain Intel chipsets (yet) and will never work on Chromebooks with ARM architecture.

This rules out the exclusive installation of Linux on systems with Samsung's Exynos processors or Nvidia Tegra chips. Also CPUs by vendor Rockchip remain sidelined. Thus, it's advisable to check the lists with the basic technical specifications of the devices on the Internet before buying a Chromebook [3].

Software

For this test, my choice of distribution for the Chromebook was GalliumOS [4]. It is based on Xubuntu but is specifically optimized for the Intel hardware in some Chromebooks. Different versions of the system are available for different Intel processor generations; it currently supports the second generation (Sandy Bridge) architectures through the fifth generation (Broadwell).

Additionally, an ISO image for the Bay Trail architecture by Intel is also available, which groups the latest Atom processors. The ISO images each install a 64-bit operating system and are optimized in the form of special adaptations to the kernel especially for the less powerful hardware of the Bay Trail architecture. For example,

Lead Image © Victoria Moody, 123RF.com

two alternative schedulers for I/O operations (BFQ) and processes (BFS) are integrated into the kernel: They simplify process organization and minimize the latency when accessing memory.

The developers have also disabled unneeded modules in the kernel and removed unused system services. They have reduced the minimum GPU clock speed and improved the often unsatisfactory performance of the touchpad modules on standard systems by using the module from Chrome OS. Instead of the LightDM, LXDM – used regularly in Xubuntu – is on board as the display manager; again this conserves system resources.

GalliumOS is also very carefully documented; a compatible hardware list exists to help you choose the appropriate operating system components [5].

Hardware

To test the operating system as efficiently as possible, I used a brand-new Chromebook 14 G4 from Hewlett-Packard and a Chromebook 11 G4 EE by the same manufacturer (Table 1).

The current hardware let us check whether corresponding drivers are available on GalliumOS and what kind of resources the less powerful hardware offers. After all, the Chromebook 14 is equipped with a Celeron type N2940 processor and is thus a latest generation Bay Trail processor, offering quad-core performance without hyperthreading [6].

The Chromebook 11 G4's Celeron N2840, however, has only two cores, although its clock speed is slightly higher [7]. The processors which belong to the

Atom CPU family are characterized by extremely low power consumption: Under load, they need a maximum of 7.5 watts; long battery life can thus be expected. The devices also work without fans and thus silently.

Both CPUs access 4GB memory on their respective devices. Additionally, the 14-inch device offers a conventional HD display with a resolution of 1366x768 pixels, although the manufacturer alternatively offers FHD-IPS display (1920x1080 pixels resolution).

The Chromebook 11 has an 11.6-inch HD display. The dual-band WiFi adapter installed in both devices, which supports the standard 802 11ac, is brand new. The devices also provide three or two USB ports, an HDMI port, and a headphone jack. There is also a slot for SD cards [8]. In terms of mass storage, the devices have eMMC Flash modules installed, each with a capacity of 16GB. Alternatively, there are models with 32GB eMMC cards as internal storage.

Preparations

To modify the hardware, you initially switch the device to root or developer mode. To do this, press Esc + F3, and at

Figure 1: The slightly modified keyboard layout of the Chromebook requires a little getting used to.

the same time switch on the device. Chromebooks have a slightly modified keyboard without dedicated F-keys, you need to take into account that the Refresh button on the Google device is the F3 key (Figure 1).

After a warning message telling you that Chrome OS is supposedly missing, which you can skip by pressing Ctrl + D, the operating system displays a mask for logging on. You need to register with Google here and then sign up with the access data of your new Google account.

If you are already registered with Google, you can log in with the existing data to authenticate. The device is now in developer mode, and you can switch it off again.

The next step is to remove the read-only protection for the firmware, which requires opening the unit. Hewlett-Packard provides for all its devices, including the Chromebook, detailed documentation in the form of service guide PDF file [9]. Because both devices have BIOS modules that allow writing and both are protected by a screw, you need to remove the screw to modify the firmware (Figure 2).

You should strictly follow the instructions to avoid accidentally damaging other components. The screws are marked by a large triangle printed on the circuit board next to them and have a flat, oversized head. After removing the write-protect screw, reassemble the device; no further hardware intervention is needed. Then reboot and skip the messages again by pressing Ctrl + D.

After signing on and enabling the Internet connection by entering the data for the purpose of authentication, press

TABLE 1: Test Equipment	HP Chromebook 14 G4	HP Chromebook 11 G4 EE
Processor	Intel Celeron N2940	Intel Celeron N2840
Number of cores	4	2
Frequency	1.83 to 2.25 GHz	2.16 to 2.58 GHz
Memory	4GB DDR3L SDRAM	4GB DDR3L SDRAM
Mass storage	16 or 32GB eMMC	16 or 32GB eMMC
Display size	14-inch LED screen	11.6-inch (LED or IPS screen)
Resolution	1366x768 or 1920x1080 pixels	1366x768 pixels
Connections	1 USB 3.0, 2 USB 2.0, HDMI, audio, network	1 USB 3.0, 1 USB 2.0, HDMI, audio, network
Communication	WLAN 802 11ac, Bluetooth 4.0	WLAN dual-band 802.11a/b/g/n/ac, Bluetooth 4.0, optionally UMTS/HSPA/LTE
Slots	SD card slot for SD/SDHC/SDXC/MicroSD cards	SD card slot for SD/SDHC/SDXC/MicroSD cards
Price	From $359 (with FHD display)	From $169
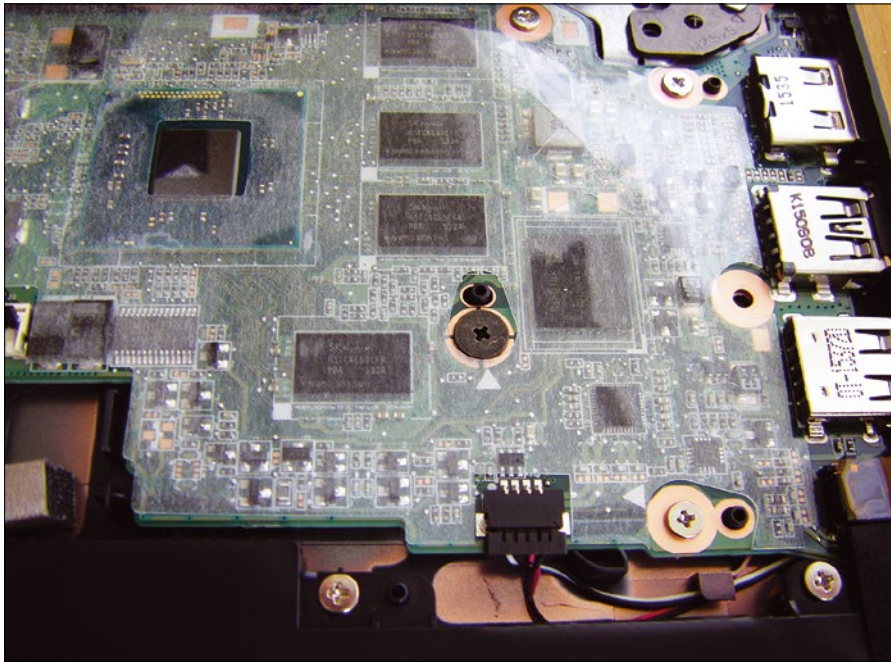
Figure 2: The read-only screw for the firmware is clearly marked in the HP Chromebook.

Ctrl + Alt + T to pop up a terminal. Enter the `shell` command at the Crosh Developer prompt to launch bash. Subsequently, enter the command sequence from Listing 1.

This loads the alternative firmware from the specified website and installs it. When replacing the firmware, follow the messages on the screen precisely, because incorrect input will destroy the firmware and lead to an unusable system.

The routine installs the free Coreboot code, to start the mainboard and all its components. Then, the script loads the SeaBIOS as payload for Coreboot. Because these operations are transparent, no more manual settings are necessary. After successfully replacing the firmware on the device, turn the machine off.

Preparing a USB Stick
Then, download the ISO image from GalliumOS from the project site and install it on a USB stick. This needs to be a device with at least 4GB capacity; you should delete all partitions before installing the ISO image. First, type the `lsblk` command to discover the USB flash drive's name.

Remove any existing partitions on the drive by typing the `fdisk/dev/<device file>` or `cfdisk/dev/<device file>`, taking care to replace the placeholder with the correct name in each case. You need to enter both commands with administrator rights.

After following the steps to delete all partitions, unmount the stick. Now copy the ISO image from GalliumOS to the storage medium by typing the command `dd if=<Image-Datei> of=/dev/ bs=1M`. The process can take up to several minutes depending on the throughput of the system.

When done, type the `sync` command to write all the data from memory to the device. Then, remove the stick from your computer and plug it into the USB port on the Chromebook.

First Start
When you first start the Chromebook with the USB stick plugged, the host calls the Sea-BIOS and branches to the GRUB boot manager on the

Flash stick, in which you can launch GalliumOS Live. The second option in the Boot Manager *GalliumOS CLI* boots the live version of the operating system up to a login prompt but does not enable the desktop.

The option *GalliumOS Live Image and Installer* booted on the modified HP Chromebook without any trouble and very quickly showed us a live system with a visually customized Xfce desktop (Figure 3).
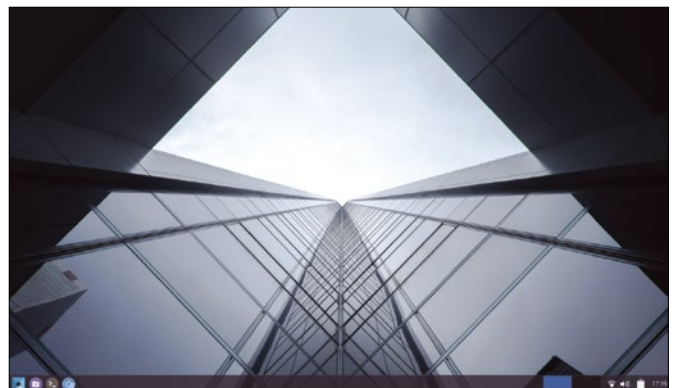
The Operating System
The distribution already has a surprisingly extensive inventory of software in the live version, despite the small size of the ISO image of less than 900MB. You will find all the typical submenus and a wide variety of applications in every menu.

Because of the small size, the live system does without programs that hog scarce resources: Instead of LibreOffice, the *Office* submenu offers you Gnumeric and AbiWord as programs for your digital office, a Chromium web browser is installed, and there is no tool for editing images.

There are just a couple of viewers. All-rounder VLC is integrated into the system for viewing multimedia content, however, and the *Accessories* menu offers a useful number of utilities.

To install the operating system on the mass storage device, double-click the *Install GalliumOS install* icon on the desktop to launch the graphic routine known from Ubuntu, which guides you through the install in just a few steps, giving you a stationary OS on the Chromebook. The installer chooses the `mmcblk0` eMMC mass storage device as the installation location; these computers have no conventional hard disk or SSD (Figure 4).

Figure 3: Virtually indistinguishable from a conventional Xubuntu: GalliumOS on the Chromebook.

LISTING 1: Load Alternative Firmware

```
$ cd
$ rm -f flash_chromebook_rom.sh
$ curl -L -O https://johnlewis.ie/flash_chromebook_rom.sh
$ sudo -E bash flash_chromebook_rom.sh
```

After a few minutes, the system is ready for use. After a reboot, it is advisable initially to set up access to the Internet. To do this, right-click the WiFi icon in the Panel and then enter the data to authenticate against the desired network.

In the *System* menu, *Synaptic Package Manager* takes you to the graphical front end for the package manager where you have access to the complete inventory of Canonical's operating system due to the compatibility of GalliumOS with Ubuntu. That is just less than 50,000 packages as Synaptic will tell you. By default, all repositories are already unlocked.

Hands-on Test

For both HP Chromebooks, the first task after the successful installation was testing the resource requirements of the operating system. It turned out that the need for memory and mass storage is astonishingly low: In spite of the now somewhat larger Xfce desktop, it only required about 420MB RAM. Also, the entire operating system occupies only around 3.5GB on the mass storage.

These very low values show you how carefully the creators of GalliumOS have integrated both the lean application software with the system and also that they have worked very hard on the kernel to speed up the system. In fact, the weaker two-core Celeron processor in the smaller Chromebook showed hardly any load (Figure 5).

The excellent hardware support is striking – despite the fact that this is a still very recent chipset with a brand-new WiFi card in the lab machine. Even the audio components of the Chromebook ran without trouble.

Very significant improvements compared to previous versions of Ubuntu were also visible in the support for the various operating modes of the mobile computer: The device woke up from sleep mode without fail after closing and then opening the lid.

Other Advanced Configuration and Power Interface (ACPI) problem areas exist, such as the time-controlled dimming and switching off the screen after a defined period of inactivity or the WiFi functionality after a suspend worked perfectly.

To test the performance limits of the Chromebooks with Linux, I installed a software tool for transcoding videos on both devices in another test: The Handbrake program not only requires more memory than conventional office or Internet applications but makes use of various hardware extensions of the Intel architecture.

For example, Handbrake uses the SSE4 instruction set introduced to Intel-based processors in the year 2007 to accelerate streaming of multimedia content. The CPU I used does not have these extensions; the conversion thus ran at a significantly slower speed.

Measurements on the devices with a standard Phase Alternating Line (PAL) resolution of the videos

Figure 4: The mass storage device in the Chromebook goes by the name of `mmcblk0`.

showed conversion speeds of between 35 and 60 frames per second – which is approximately the level of Core-2 Duo processors of the Merom and Penryn generations (Figure 6). In other words, the HP laptops can easily take on full-blown laptops of a slightly older generation.

Another positive thing in this context is that the test equipment generated little waste heat because of the low energy consumption, even at full load. There

was only a slight increase in temperature directly below the CPU on the housing.

And, the battery life was not noticeably reduced, even under full load: Although an external DVD drive connected via a USB port served as the source for transcoding the videos, the estimated run time was around five hours.

Even extensive office and Internet applications, such as LibreOffice and Firefox, which also ran on the HP ma-

chines in our lab, failed to faze the systems. The systems also managed to stream HD videos from the Internet in full-screen mode, although the CPU load on both cores of the N2840 processor rose significantly up to about 80 percent.

I also enjoyed the very well-designed audio system when playing back multimedia content: GalliumOS had no trouble addressing the audio system of the Chromebook correctly, and, thanks to a sophisticated speaker system on the bottom side of the unit, it offered an unusually full and clear sound for a mobile computer.

Conclusion

As the test impressively demonstrated, Chromebooks are definitely suitable for more than just a few web applications and, thanks to the free developer community, you rid many of these devices of those shackles installed by Google.

GalliumOS, which boasts excellent hardware support even for very recent components, turns the low-cost laptops into full-fledged systems for your daily needs.

Because the two test devices from Hewlett-Packard also came with good displays, a Chicklet keyboard that was nice to use, and excellent housing quality, they prove to be serious competitors to traditional laptops, while at the same time taking the strain off your budget. ∎∎∎

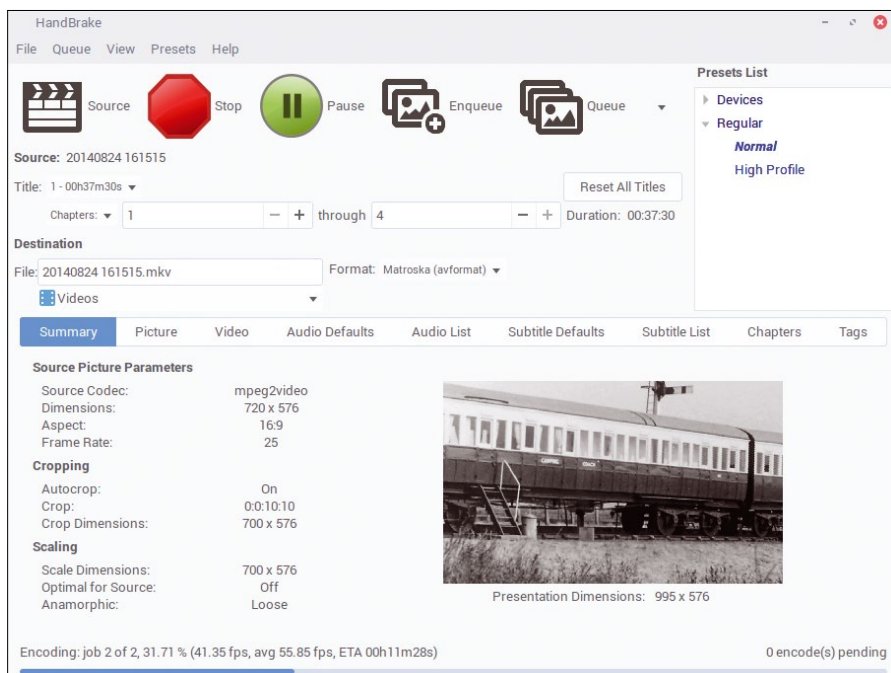**Figure 5:** System load on GalliumOS GDP.

Figure 6: Almost as quick as large notebooks, the Chromebooks can handle even computationally intensive tasks.

▮ INFO

[1] Chrx: *https://chrx.org*

[2] Crouton: *https://github.com/dnschneid/crouton*

[3] Overview of Chromebooks: *http://www.chromebookspecs.com/*

[4] GalliumOS: *https://galliumos.org*

[5] Hardware compatibility list: *https://wiki.galliumos.org/Hardware_Compatibility*

[6] Technical specification: *http://ark.intel.com/products/82104/Intel-Celeron-Processor-N2940-2M-Cache-up-to-2_25-GHz*

[7] Technical data: *http://ark.intel.com/products/82103/Intel-Celeron-Processor-N2840-1M-Cache-up-to-2_58-GHz*

[8] Technical specification HP Chromebook 14 G4: *http://www8.hp.com/de/de/products/laptops/product-detail.html?oid=9740875&jumpid=oc_r1002_dede_c-001_r0002#!tab=specs*

[9] Download manual: *http://h10032.www1.hp.com/ctg/Manual/c04823664*

IT Highlights at a Glance

Too busy to wade through press releases and chatty tech news sites? Let us deliver the most relevant news, technical articles, and tool tips – straight to your in box. Subscribe today for our excellent newsletters:

- ADMIN HPC
- ADMIN Update
- Linux Update
- Raspberry Pi

and keep your finger on the pulse of the IT industry.

Admin and HPC: www.admin-magazine.com/newsletter
Linux Update: www.linuxpromagazine.com/mc/subscribe
Raspberry Pi: www.raspberry-pi-geek.com/mc/subscribe

©Yuri Arcurs, 123RF.com

The sys admin's daily grind: The Fuck

Expletives Not Deleted

In the 25 years of Linux kernel history, a huge amount of code and many comments have accumulated. They can be explanatory, enlightening, funny, or even profane. Charly prefers to do things differently, so he types *fuck* in the shell. *By Charly Kühnast*

… profanity provides a relief denied even to prayer. – Mark Twain

As early as 1998, Sean Dreilinger created the first statistics on swear words in Linux [1]. It showed evidence of a sharp increase of the word "fuck" in kernel 2.1.5. Vidar Holen [2] delved even deeper into the murky depths (Figure 1). He found evidence of almost 50 incidences of "fuck," many incidences of "shit," and even 180 incidences of "bastard."

How often Linus has raised a warning middle finger against hardware vendors such as Nvidia is unfortunately unknown; however, contemporary historians have probably only investigated the kernel mailing list quantitatively.

Damned Shell!

Cursing has made its way out of the kernel to other system components; a good example of this in many ways is The Fuck [3], a semi-automatic correction function for shell input. The tool, written in Python 3 is installed as follows:

```
sudo apt install python3 python3-dev
git clone https://github.com/nvbn/thefuck
cd thefuck
sudo ./setup.py install
```

After installing, if you make a typing error, such as forgetting the space in

cd .., you first see the standard *-bash: cd..: command not found*. However, if you then type `fuck`, The Fuck suggests the correct command,

```
cd .. [enter/^/V/ctrl+c]
```

and you just need to confirm by pressing Enter. Of course, `cd..` is a very simple example. Many users will already have created aliases for these typical errors. Incidentally, my favorite alias is

```
alias doch='sudo $(history -p !-1)'
```

where *doch*, in this case, roughly translated into English, could mean "actually" or "of course" (slap forehead). This alias

saves me much suppressed cursing after forgetting to `sudo`.

However, The Fuck can do far more; it can even prevent you from shooting yourself in the foot with more complex Git commands. Ergo, the world needs much more of The Fuck to make it a more polite place. ■■■

█ INFO

[1] Linux kernel fuck count: *http://durak.org/sean/pubs/kfc/*

[2] Linux kernel swear counts: *https://www.vidarholen.net/contents/wordcount/*

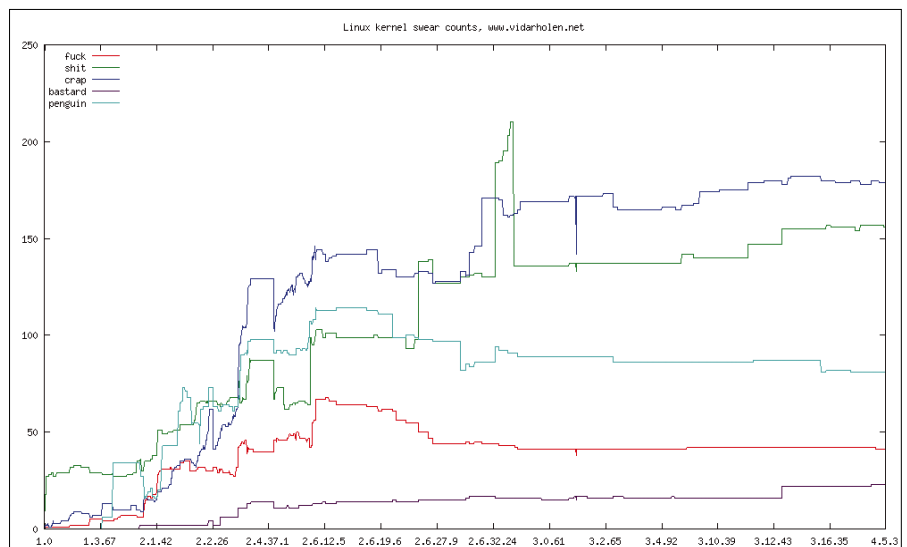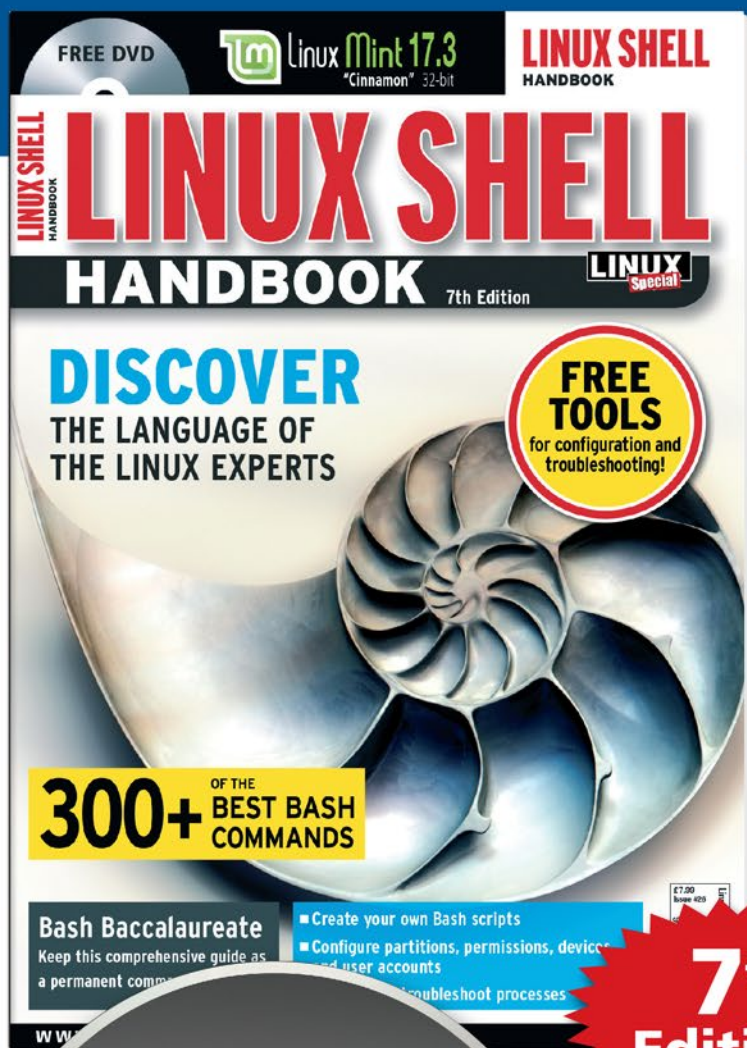[3] The Fuck: *https://github.com/nvbn/thefuck*

Figure 1: Kernel versions are shown on the *x* axis, and incidence of words on the *y* axis. Top to bottom: crap (blue line), shit (green), penguin (teal), fuck (red), bastard (violet).

█ CHARLY KÜHNAST

Charly Kühnast is a Unix operating system administrator at the Data Center in Moers, Germany. His tasks include firewall and DMZ security and availability. He divides his leisure time into hot, wet, and eastern sectors, where he enjoys cooking, freshwater aquariums, and learning Japanese, respectively.

Perl 6 in a hands-on test

Double Down

Perl 6 gives the Perl 5 syntax a new paint job and introduces a genuine object system. Mike gives it a hands-on test based on a re-implementation of the old blackjack game. *By Mike Schilli*

You should never write off the old dogs: After decades of anticipation, Perl 6 finally seems to be taking shape. I found this out recently when I attended the grass roots Perl Conference YAPC 2016 in Orlando, Florida. About two-thirds of the talks were dedicated to Perl 5 topics, but to my amazement, more and more people in the community are seriously taking to the successor and problem child of many years standing, Perl 6.

No less than 11 years ago at the OSCON Conference 2005, Damian Conway seriously announced that the "endgame" of the development had started, but multiple setbacks and new beginnings ensued. Then, suddenly, at the end of 2015, the Rakudo virtual machine finally become operational, and although it does not exactly impress with its performance, it does at least reliably parse Perl 6 code.

What Is Perl 6?

To find out what you need Perl 6 for, it is helpful to analyze what Perl 5 lacked. Shortcoming number one was without a doubt a full-blown object system. In Perl 5, it was a makeshift affair, and no one in their right mind would have used the blessed hash structures, dubbed classes, without additional strictures in anything but toy-sized systems.

Instead, for the past 10 years, professional Perl developers have used the CPAN module Moose or one of its offshoots to define classes and access their attributes. This approach not only removes the need for developers to type tedious boilerplate code but also establishes a safety net, because the Perl engine immediately notices if a typing mistake has crept in. This is not the case if you have an incorrect hash key, resulting in undefined values being returned and random programmatic outcomes that are destined to creep up on you unawares. Any other modern language has solved this problem, but Perl 5 unfortunately only has a patchwork implementation for historic reasons.

Noisy Code

The makers, Larry Wall and Damian Conway, thus gave Perl 6 syntax for defining classes and invoking object methods that can take lists of named parameters. While they were at it, they adjusted some logical inconsistencies in the Perl 5 syntax that had proved to be pitfalls for beginners.

They didn't stop there, however; they kept going by borrowing syntax from various modern languages such as Python or Ruby, with the promise of practical use and clever programming tricks. The result is a language with a relatively large number of special characters that completely breaks compatibility with the old Perl 5 and needs a new interpreter named Rakudo.

Schrödinger's Cat

One of the smart and exotic new constructs in the new version involves superpositions, called "junctions" in Perl 6 and penned by Damian Conway. Conway peddled the idea 13 years ago at conferences, and later introduced it to the language in the form of quantum variables as a CPAN module by the name of Quantum::Superpositions.

Like the famous cat in Schrödinger's experiment [1] that is simultaneously dead and alive, these variables can simultaneously assume several values. For example, an integer can be 1 and 2 at the same time, and adding 5, gives you a result of 6 and 7. Back in the day, while

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*.

trying to come up with a practical example for this, I remembered the casino game blackjack as an example, where an ace counts as either 1 or 11.

Blackjack in Perl 6

So, in 2004, I wrote a column with a script that simulated a blackjack game for *Linux Magazine* [2] using superpositions to help count the value(s) of a hand of cards. This month, with my newly acquired Perl 6 knowledge, I thought it could be worthwhile to try to implement the same blackjack program in modern Perl 6.

Listing 1 [3] shows how Perl 6 implements the class code for a single blackjack card from the deck. Cards are valued by their ranks; the Blackjack::Card class thus has an attribute named rank in Listing 1 with a Perl scalar as the default type capable of accommodating a string or an integer. If you have worked in Perl 5 with the Moose module, you do not need to change all that much for Perl 6; the new syntax is almost identical.

In addition to a suit, each card also has a count value val and a color (suit_color) when displayed on the screen. The definitions in Listing 1 begin with a dollar sign followed by a dot, which distinguishes these attributes as a Perl scalars. Perl 6 also offers arrays (@) and hashes (%) for attributes, as well as arbitrary class names. The dot after the "sigils," as these special characters are known, distinguishes a variable as an object attribute.

Perl 6 defines handy methods for read and write access behind the scenes at the same time (for example, obj.rank("A") to

Figure 1: The blackjack dealer has dealt himself a face-up and a face-down card and each of the six players at the table two face-up cards. Image © Tomas Hajek, 123rf

set the value) and named parameter lists for the constructor, which then understands calls such as Blackjack::Card.new(rank => "A"). Important: Perl 6 replaces the arrow notation -> by a dot; $o->foo() in Perl 5 thus becomes $o.foo() in Perl 6. The self keyword (note the missing sigil) is used to access an object within the object itself.

Strings in Perl 6 can include not only automatically interpolated variables, such as "Hello $guest", but also dynamically executed code constructs in curly brackets. This is a big help, especially with the new object syntax if the value does not exist in a variable but originates from a method: "Hello {$o.guest()}" hence replaces the curly brackets and everything in between with the return value of the guest() method for an object that resides in $o.

Blackjack 101

How is blackjack played in a casino in Las Vegas? At a card table, the dealer stands facing the players and deals cards from a 52-card deck in a shoe (Figure 1). To make life harder for the card counters among the players in the casino, a shoe contains up to eight shuffled decks of 52 cards.

First, the dealer deals two cards to each player at the table, and then deals him/herself one face-up and one face-down card.

Then it is the players' turn; a player can either ask for more cards from the dealer ("hit") or politely refuse ("stand"). The goal of the game is to get as close as possible to the magic number of 21 with all drawn cards – but without exceeding this number – because the player otherwise automatically loses ("bust"). Once all players are satisfied, the dealer reveals the face-down card and may possibly draw more cards from the shoe.

Soft and Hard 17

Like robots, dealers in Las Vegas play a fixed system: They draw new cards until they have at least 17 points. For example, if a 10 comes out of the shoe, followed by a six, the dealer draws again, even though this will typically mean overshooting the target 21 and going bust, although an opponent may have, for example, already stopped at 13 to avoid drawing a 10 and going bust.

This rule is identical for all casinos, with a minor variation, the so-called "soft 17" [4]. This rule determines whether a dealer with a hand of 17, including at least one ace (which counts as either 1 or 11), needs to draw again. If the dealer draws, for example, an ace and a six, they have 17 and would no

LISTING 1: Card.pm

```
01 use v6;
02 use strict;
03 use Terminal::ANSIColor;
04
05 class Blackjack::Card {
06   has $.rank;
07   has $.suit;
08   has $.val;
09   has $.suit_color;
10
11   method Str() {
12     return colored(
13       "{self.rank}", self.suit_color );
14   }
15 }
```

longer draw a card following the "hard 17" rule. If the casino uses the "soft 17" rule, however, the dealer must continue, because this 17 (if the ace is counted as 1 instead) can also be interpreted as 1 + 6 = 7. Statistically, this gives the dealer a small advantage; their chances of winning grow by 0.2 percent.

Blackjack in the Terminal

In the command-line version of the script featured in Figure 2, blackjack.p6, the dealer follows the "soft 17" rule and plays against one other player. Figure 2 shows a game in which the player wins against the dealer. First, the dealer gets a Q (for queen) and a face-down card, which the script does not display. The player receives two 2s and types h ("hit"), to ask for one more card. It is an ace (A), and given that an ace in blackjack is either 1 or 11, the value of the hand of cards is now 5 or 15, which the script shows as any(5,15).

Another hit gives the player a 10; the hand of cards is either 15 or 25, although the latter gets discarded being over 21. The player boldly presses h again, receives a 6 as if by a miracle, and now has precisely 21, as the alternative count of 31 makes no sense. The player presses s

("stand"), and it's the dealer's turn. The dealer turns over the face-down card (a 4), so that he or she now has Q-4, a queen and a 4, which makes 14 all told. Because 14 is less than 17, the dealer draws one more card in accordance with the rules of the casino, but gets a jack (J) and has now gone bust with 27: game over. The player wins the point (score: 1), and his total score (total) is now 1 as well, because this was the first game.

Cards in a Poker Hand

The playing cards in a 52-card hand are represented by a Blackjack::Deck class object in Listing 2. So that the Perl parser knows that Perl 6 code follows and won't choke on the new syntax, the file starts with use v6: This defines the version in a syntax that the Perl 5 interpreter also understands causing

it to abort with an error message should someone accidentally point Perl 5 at the program. Line 7 uses the following expression

```
flat < A J Q K >, 2..10
```

to define a list of possible values of cards (ace, ten, jack, queen, king), as well as the numbers from 2 to 10.

Space-separated values in angle brackets are the Perl 6 syntax for the qw(...)

LISTING 2: Deck.pm

```
01 use v6;
02 use strict;
03 use Blackjack::Card;
04
05 class Blackjack::Deck {
06   has @.ranks =
07     [ flat < A J Q K >, 2..10 ];
08   has @.suits =
09     < Hearts Spades Diamonds Clubs >;
10   has %.suit_color =
11     flat self.suits Z
12     flat < red black magenta blue >;
13   has @.cards;
14
15   method shuffle() {
16     for self.ranks -> $rank {
17
18       my $val = 10;
19       if $rank ~~ /\d+/ {
20         $val = $rank.Int();
21       } elsif $rank eq "A" {
22         $val = 1|11;
23       }
24
25       for self.suits -> $suit {
26         self.cards.push(
27           Blackjack::Card.new(:$rank,
28             :$suit, :$val,
29             suit_color =>
30             self.suit_color{ $suit } ));
31       }
32     }
33     self.cards = self.cards.pick( * );
34   }
35
36   method pick() {
37     if self.cards.elems == 0 {
38       self.shuffle;
39     }
40     return self.cards.shift;
41   }
42 }
```

```
$ ./blackjack.p6
Dealer: Q (10)
Player: 2-2 (4)
Your move (h/s/q):

Player: 2-2-A (any(5, 15))
Your move (h/s/q):

Player: 2-2-A-10 (any(15, 25))
Your move (h/s/q):

Player: 2-2-A-10-6 (any(21, 31))
Your move (h/s/q):

Dealer: Q-4 (14)
Player: 2-2-A-10-6 (any(21, 31))
Dealer: Q-4-J (BUSTED)
Score: 1
Total: 1
```

Figure 2: The blackjack game in action: Here the casino player wins against the dealer.

lists in Perl 5. The range operator `..` existed in Perl 5 too, except it returned a long list of values, whereas Perl 6 returns a single item that contains that list. Therefore, the expression above needs an additional `flat` operator to flatten out the whole thing and create a long list.

To simulate all the cards, the `shuffle()` method in line 15 of Listing 2 iterates through the card attributes in two for loops, permutating `@.ranks` and `@.suits`, and combining two entries for each card of the Blackjack::Card class. The code then pushes the object created in this way onto the `@.cards` array. Lines 18-23 determine the count value for each card, the 1 | 11 entry for an ace is one of the previously discussed superpositions (called "junctions") of the values 1 and 11.

Zipper

The screen color assigned to the card colors by the `%.suit_color` hash is determined by a mapping between `@.suits`

and a flattened list of display colors in line 12. The `Z` operator at the end of line 11 applies the zip function to the two lists and assigns a screen color to each suit in the hash. The for loops in Perl 6 are syntactically different from loops in Perl 5; the `for` keyword is followed by a list or an array and then an arrow operator and the name of the variable to assume the current iteration value within the loop block.

To mix the deck, line 33 uses a clever trick involving the so-called "whatever" operator `*`. The `pick()` method selects a random element from an array, removes it from the array, and returns the element. The `*` tells Perl 6 in this construct to continue until the array is empty and then shuffles the original array and returns it as a list. The values are then assigned to the variable on the left side of the equation. A user-defined method of same name defined at line 36 removes a card from the mixed deck and returns it to the caller.

Points with Quantum

Listing 3 implements the Blackjack::Hand class. The `draw()` method in line 9 expects an object of the Blackjack:Card type and adds it to the player's hand. To determine the count value (or rather the superimposed counts) in the hand, the `values()` method in line 20 counts numerical values of all the cards. The critical value of 21 is exceeded if none of the superimposed states shows 21 or less, and the method `is_busted()` in line 28 returns a value true in this case.

To display the player's hand, all you need to do is put a Blackjack::Hand class object in a string context (i.e., in double quotes). This tells Perl 6 to invoke the object's `Str()` method and use its return value. To allow this to happen, the function in line 32 of Listing 3 uses the `cards()` method to iterate through all the cards in the player's hand, puts each card object in a string context in line 36, and joins the results using dashes as separators.

LISTING 3: Hand.pm

```
01 use v6;
02 use strict;
03 use Blackjack::Card;
04
05 class Blackjack::Hand {
06   has @.cards;
07   has $.name;
08
09   method draw( Blackjack::Card $card ) {
10     push self.cards, $card;
11   }
12
13   method highval() {
14     my $vals =
15       grep { $_ <= 21 }, self.values;
16     return "BUSTED" if !$vals;
17     return $vals;
18   }
19
20   method values() {
21     my $total = 0;
22     for self.cards -> $card {
23       $total += $card.val;
24     }
25     return $total;
26   }
27
28   method is_busted() {
29     return !( self.values <= 21 );
30   }
31
32   method Str() {
33     my $str = "";
34     for self.cards -> $card {
35       $str ~= "-" if $str.chars;
36       $str ~= "$card";
37     }
38     return "{self.name}: " ~
39       "$str ({self.highval})";
40   }
41
42   method is_blackjack() {
43     return self.cards.elems == 2 &&
44     self.values == 21;
45   }
46
47   method score( Blackjack::Hand $other ) {
48     return -1   if self.is_busted;
49     return  1   if $other.is_busted;
50     return  0   if self.is_blackjack and
51       $other.is_blackjack;
52     return  1.5 if self.is_blackjack;
53     return -1   if $other.is_blackjack;
54     self.highval > $other.highval ??
55         return 1 !!
56         return 0;
57   }
58 }
```

The string returned in line 38 contains the name of the player, the cards in the hand, and the score determined by the `highval()` method. The operator for joining two strings mutated from a dot (`.`) in Perl 5 to a tilde (`~`) in Perl 6. Line 36 thus uses `~ =` to append the string on the right to the string on the left.

Jackpot with Blackjack

If a player is holding a card with a value of 10 along with an ace, this counts as a blackjack. If the dealer does not have the same combination, the player is paid 1.5 times their bet as winnings. The `is_blackjack()` method tests for this combination as of line 42, by ensuring that the hand consists of exactly two cards and that one of the superposed states scores exactly 21.

The winning points earned by the player are determined by the `score()` method in line 47, which expects the dealer's hand object as an argument for the comparison. The return value is `-1` if the player has gone bust, no matter what the dealer has. If the dealer goes bust, the return value is `1`, and the player is one point ahead. If both have a blackjack, the result is a tie and `0` is returned. Note that the ternary operator in Perl 5, using the syntax `[...]?[...]:[...]` to return the first or second expression depending on the test, has mutated in Perl 6 to `[...]??[...]!![...]`.

Command-Line Casino

The controlling Perl script of this command-line casino `blackjack.p6` is shown in Listing 4; its first line contains a reference to the executing Perl 6 interpreter. But where do you find Perl 6? On *perl6.org*, you will find a tarball which you can build yourself while hundreds of warnings flash across the screen. The easiest approach for a modern Linux enthusiast is to run `docker pull rakudo-star` to download a Docker image with Perl 6 Rakudo installed.

To call a Perl 6 script residing on the host in the Perl 6 interpreter within the Docker container, Listing 4 defines the path to the shell script in Listing 5 in its first line, or shebang. This starts the Docker container and defines a mount in the current directory on the host, which is then visible within the container below `/perl6`. This is why line 4 of Listing 4 adds an additional search path for modules in the form of `use lib '/perl6'`

LISTING 4: blackjack.p6

```
01 #!/usr/bin/env perl6-in-docker.sh
02
03 use v6;
04 use lib '/perl6';
05 use Blackjack::Card;
06 use Blackjack::Hand;
07 use Blackjack::Deck;
08
09 my $TTY   = open("/dev/tty");
10 my $deck  = Blackjack::Deck.new;
11 my $total = 0;
12
13 while ( 1 ) {
14   my $dealer =
15    Blackjack::Hand.new( name => "Dealer" );
16   my $player =
17    Blackjack::Hand.new( name => "Player" );
18
19   $player.draw( $deck.pick );
20   $player.draw( $deck.pick );
21
22   $dealer.draw( $deck.pick );
23   say "$dealer";
24   $dealer.draw( $deck.pick );
25
26   while !$player.is_busted {
27     say "$player";
28     my $in =
29       prompt-char("Your move (h/s/q):");
30     say "";
31     given ( $in ) {
32       when 'q' { exit( 0 ); }
```

```
33       when 's' { last; }
34       when 'h' {
35         $player.draw( $deck.pick ); }
36     }
37   }
38
39   while !$player.is_busted and
40         !$dealer.is_busted and
41         $dealer.values < 17 {
42     say "$dealer";
43     $dealer.draw( $deck.pick );
44   }
45
46   say "$player";
47   say "$dealer";
48
49   my $score = $player.score( $dealer );
50   $total += $score;
51
52   say "Score: $score";
53   say "Total: $total\n\n";
54 }
55
56 sub prompt-char($prompt) {
57   ENTER shell "stty raw -echo min 1";
58   LEAVE shell "stty sane";
59
60   print $prompt;
61   my $in = $TTY.read(1).decode;
62   say "\r";
63   return $in;
64 }
```

so that it also finds the blackjack modules from the other listings.

Because the Linux kernel does not allow shell scripts as shebang programs for security reasons, Listing 4 uses the helper /usr/bin/env as the executable that invokes the shell script; the Linux kernel has no trouble with this.

Game Mode

Keyboard input from the command line usually requires the user to confirm each entry with the return key, but in game mode, you want the blackjack script to respond directly to key presses. The raw terminal mode is used for this on Unix systems; you will only want to enable this during the actual entry and reset to normal cooked mode immediately after; otherwise, the terminal no longer responds to shell

LISTING 5: perl6-in-docker.sh

```
1 #!/bin/sh
2
3 sudo docker run -v `pwd`:/perl6 -i \
4    -t ready /usr/bin/perl6 /perl6/$*
```

commands typed after the program has exited.

This is why the prompt-char() function defined in line 56 (Perl 6 allows dashes in function names) uses the stty utility to run a shell command that enables raw mode and disables it again on leaving the routine. Perl 6 provides the keywords ENTER and LEAVE to perform actions when entering and exiting a function. From the $TTY file handle in the previously opened terminal, read(1) consumes a byte via the prompt-char() function, and the downstream decode() method converts the buffer byte to an ASCII character that the script can later compare with a letter such as "h" (hit) or "s" (stand).

Perl 6 has a switch statement, which goes by the name of given/when and is used in line 31 to compare the player's keyboard input with specified letters and initiate the corresponding game controls.

Perl 6 vs. Perl 5

The game flow in Listing 4 is pretty self-explanatory, and if you compare the lines of code required for the implemen-

tation, Perl 6 comes off slightly ahead. Also the dropped parentheses in the for and while loop heads have a positive effect on readability.

Will Perl 6 assert itself? It is probably too early to tell; production tests will decide the outcome. Larry Wall hinted that he was working on a Perl 6 book at the conference in Orlando but left open how long it will take to publish it. If you want to draw lessons from the past, then you might guess that a good chunk of time may pass between announcement and release in Larry's private empire. ■■■

INFO

[1] Schrödinger's cat: *https://en.wikipedia.org/wiki/Schrödinger%27s_cat*

[2] "Quantum Casino" by Michael Schilli, *Linux Magazine*, January, 2004, pg. 61: *http://perlmeister.com/lme/prod-0401.pdf*

[3] Listings for this article: *ftp://ftp.linux-magazine.com/pub/listings/magazine/191/Perl*

[4] Blackjack soft 17 rule: *http://www.readybetgo.com/blackjack/strategy/soft-17-rule-2496.html*

Mastering Kanboard

Kanban Fun

Based on the kanban system, Kanboard offers solid tools for efficiently managing projects. We explain the Kanboard basics and put the application to project management duties.

By Dmitri Popov

DMITRI POPOV

Dmitri Popov has been writing exclusively about Linux and open source software for many years, and his articles have appeared in Danish, British, US, German, Spanish, and Russian magazines and websites. Dmitri is an amateur photographer, and he writes about open source photography tools on his Scribbles and Snaps blog at *scribblesand-snaps.wordpress.com*.

The kanban technique [1] is not limited to large organizations managing complex projects. Practically any project – travel planning, writing a book, and developing software – can be managed using a kanban-style application. The basic idea behind kanban is relatively simple.

The key components of kanban are cards and a board divided into columns representing project stages. In the most simple scenario, the project may consist of three stages: To Do, Doing, and Done. Each card represents a project task, and you move cards from one project stage to another as you work on the tasks and complete them. In the case of the To Do-Doing-Done project system, you start by placing all the cards into the To Do column, then move the tasks you are currently working on to the Doing column. When the task is completed, you push it to the Done column (Figure 1).

As the name suggests, Kanboard [2] is based on the kanban technique, and the application offers a wide range of tools and features that ease the project management burden. To deploy Kanboard you need a machine running a web server (e.g., Apache) and PHP. Although the application supports several database back ends, including MariaDB and PostgreSQL, it uses the SQLite database engine by default, which requires no setup or configuration.

Installing Kanboard is straightforward. Fetch the latest version of Kanboard from the project's website, unpack the downloaded archive, rename the resulting directory to `kanboard`, and move it to the document root of the server (e.g., `/var/www`). Make the `kanboard` directory writable by the server using the

```
chown www-data:www-data -R kanboard
```

command, and you are done. Point the browser to *http://127.0.0.1/kanboard* (replace *127.0.0.1* with the actual IP address or domain name of the server running Kanboard), and log in using the default *admin/admin* username and password.

Kanboard Basics

The first step is to change the default login credentials and populate Kanboard with additional users, if necessary. To do this, expand the User menu in the upper-right corner of Kanboard's interface and choose *My Profile*. Click on the *admin* entry, then click on the *Edit Profile* item in the *Actions* section. Modify the default name and email, and press *Save*. Click on the *Change password* link and change the default password.

You might want to tweak a few other settings, too. You can upload an avatar and enable the two-factor authentication feature, which adds an additional layer of security by generating a disposable access code for logging into Kanban. To enable this feature, install the FreeOTP [3] two-factor authentication app on your mobile device, enable the
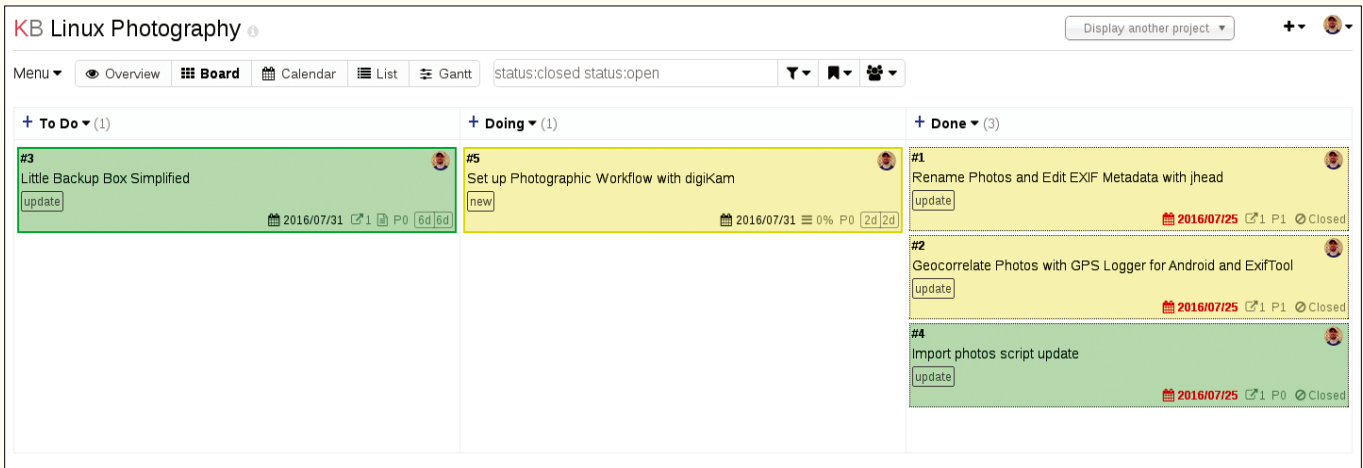
Figure 1: Kanboard allows you to manage projects visually.

two-factor authentication option for the Kanboard user, and scan the generated QR code with FreeOTP (Figure 2). Use the app to generate a code, enter it into the *Code* field, and press *Check my code* to enable two-factor authentication. To keep the user up-to-date on project activities, you can also enable and configure notifications in the appropriate section of the user management area.

Next stop is the application's settings section (choose *Settings* from your User menu). Although you can leave most options at their defaults, you might want to modify a couple of settings. In the *Application settings* section, you can change the interface language and timezone as well as specify the desired time format. By default, Kanboard uses four project columns: Backlog, Ready, Work in progress, and Done. But you can change that in the *Project settings* section by entering the desired column names (e.g., To Do, Doing, Done) into the appropriate field.

Now you are ready to create your first project. To do this, press the + button next to the User menu and select either *New project* (a project that can be accessed by other Kanboard users) or *New private project* (a project accessible only by you). Give the project a descriptive name and press *Save*. This creates the project and switches to the settings area. There are a few options you can modify here. If you prefer to use custom tags for project tasks, you can specify them in the *Tags* section.

By default, all projects in Kanboard are private, meaning the user must be logged in to the application to access projects. However, the application also allows you to make projects public (Fig-
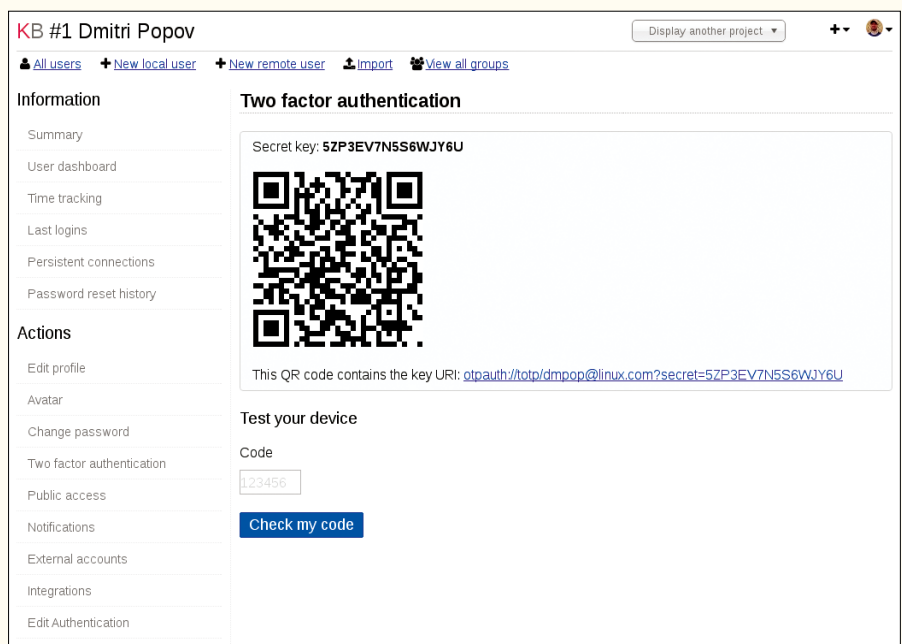
Figure 2: Kanboard supports two-factor authentication.
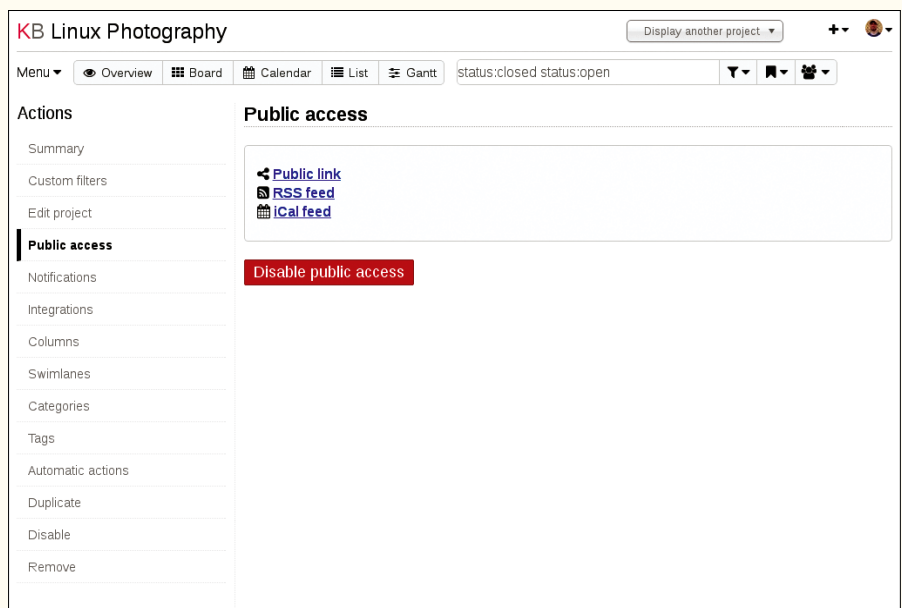
Figure 3: Making the project public.

Figure 4: Adding a new task.

ure 3). To do this, switch to the *Public access* section and press the *Enable public access* button. This not only makes the project public but also generates RSS and iCal feeds for others to subscribe to.

Before you start exploring Kanboard's interface and tools, it's a good idea to add a few tasks to the project (Figure 4). The application provides several ways to do that, but for now choose the *Add a new task* command from the main menu. Use the available fields and options in the New Task dialog to give the task a name, a description, assign tags and user, specify color, and select the project column. You can specify the task's priority and complexity as well as its start and due dates along with the estimated number of hours required to complete the task. Because Kanboard lets you manage projects visually, card colors can act as important identifiers. So, you might want to devise a system for using specific colors right from the start. For example, if you are working on a programming book project, you can choose one color for all tasks that involve coding and debugging, and another color for writing tasks. This way, you can easily identify each task by its color.

With the first project ready to go, you can start exploring Kanboard. The application's interface consists of several views. As the name suggests, the *Overview* view gives you a brief overview of the current project, including the number of tasks in each project column and a list of recent activity. *Board* is probably the most important view in Kanboard; it displays the project as a board divided into columns representing project stages.

Tasks on the board are shown as cards, and you can move them from one column to another using the mouse. You can add new tasks to a column by clicking on the

Figure 5: Kanboard features several views, including a Calendar view.
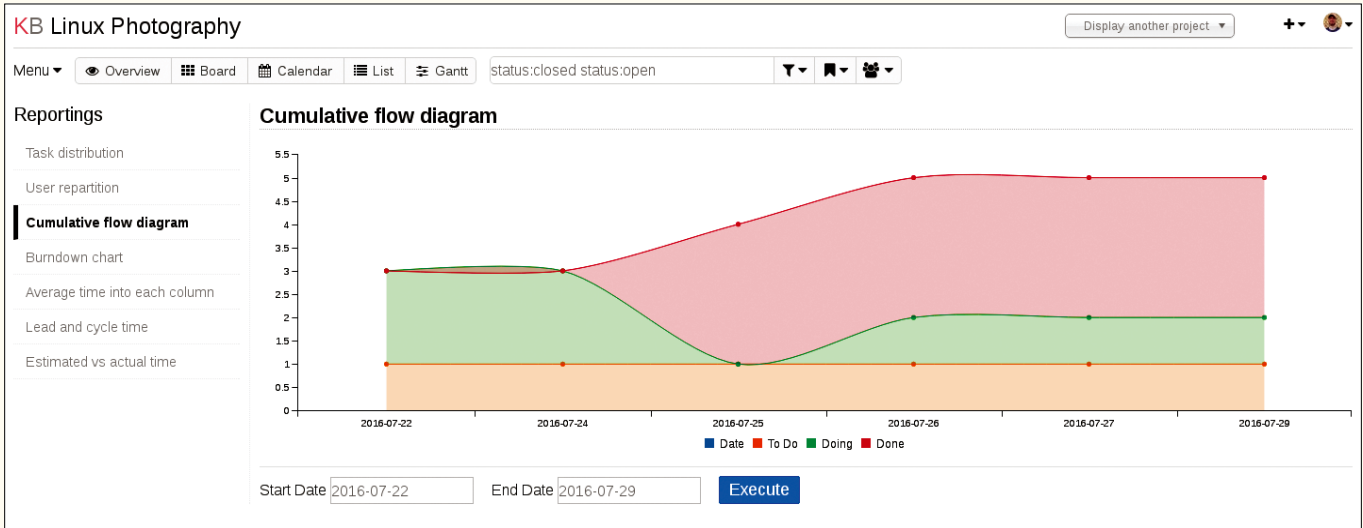
Figure 6: The Analytics area gives you access to various charts.

+ icon next to it. The *Calendar* view displays the tasks that have a due date, and the *List* view shows tasks as a regular list (Figure 5). Finally, the *Gantt* view displays the project tasks and relationships between them on a timeline (more about task relationships later). You can use the mouse to move the tasks on the timeline as well as adjust the duration and due date of each task.

Under the main menu, you'll find a handful of useful items. The *Analytics* command, for example, gives you access to the Analytics area with various charts and graphs, such as Task distribution, Cumulative flow, Average time into each column, and so on (Figure 6).

To get an overview off all your projects and tasks, switch to the Dashboard area by choosing the *My Dashboard* command from the User menu (or simply click on the KB logo in the upper-left corner of Kanboard's interface). Here, you can quickly view not only projects and tasks, but also your calendar, activities, and notifications (Figure 7). The *Exports* and *Imports* commands let you export and import tasks, and the *Activity* command shows the activity stream.

The Filter bar in Kanboard's interface is used to filter tasks by specific criteria, and the application comes with several default filtering rules like My tasks, Tasks due today, Open tasks, and the like. You also can specify queries on the fly using special syntax. For example, the *assignee:me due:tomorrow Proofread* query displays all the tasks assigned to you with a due date of the next day and containing the word Proofread in their titles. Better still, Kanboard makes it possible to create
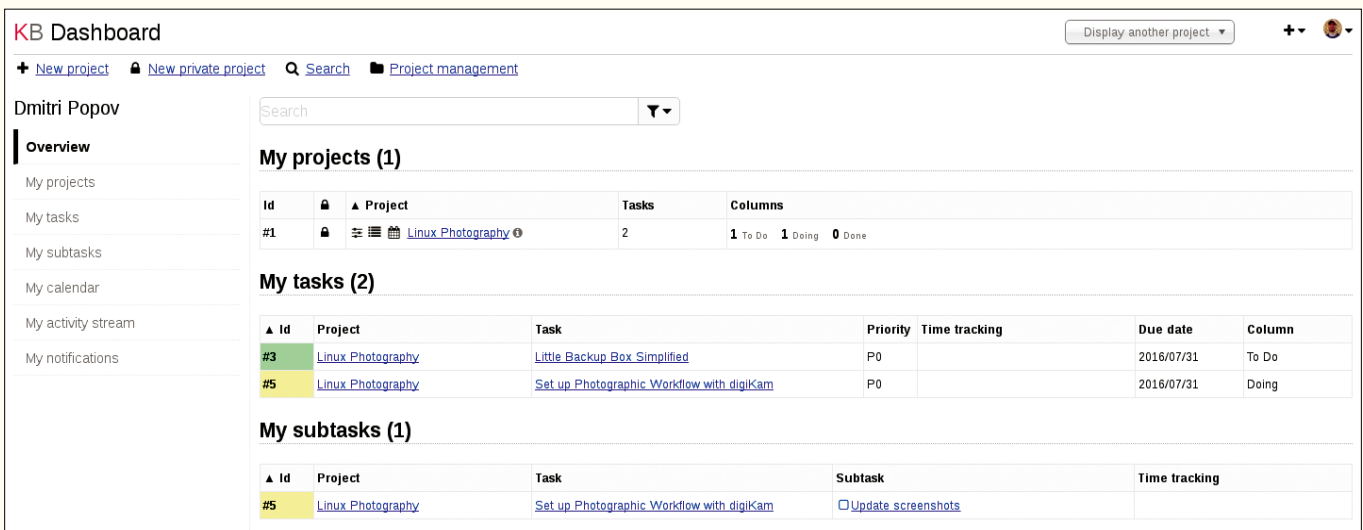
Figure 7: The Dashboard provides an overview of all your projects and tasks.
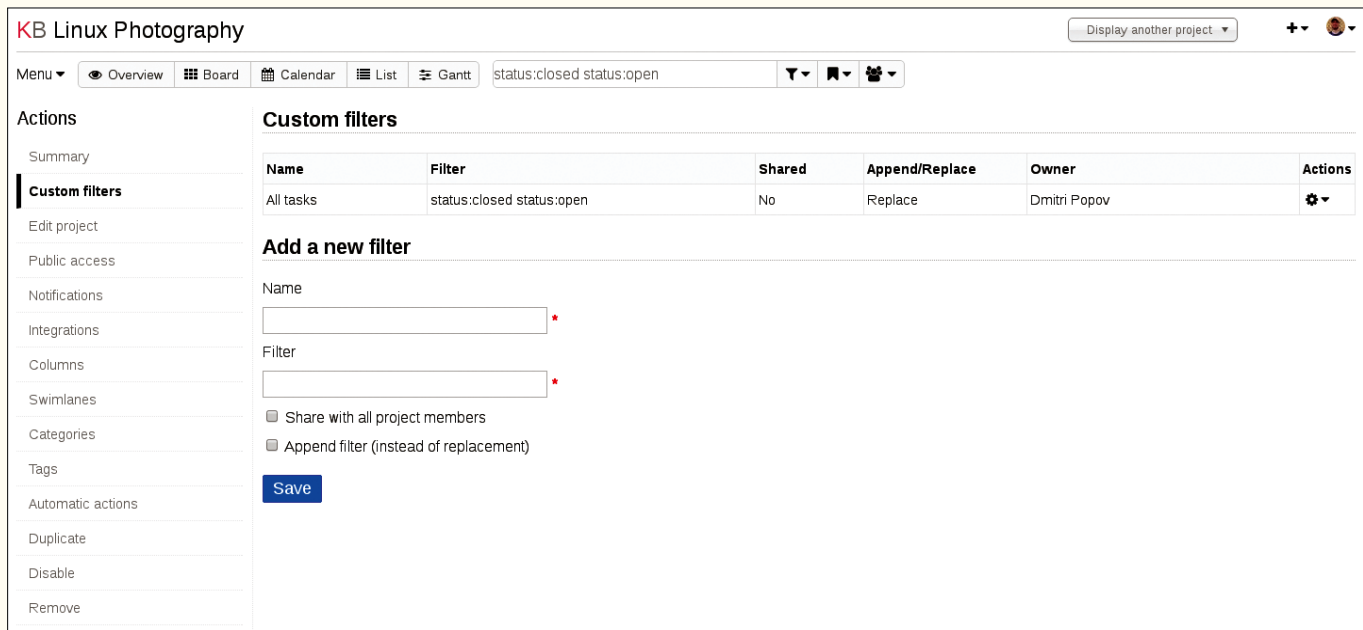
Figure 8: Kanboard makes it possible to create custom filters.

custom filters (Figure 8). To create a filter, choose *Custom filters* from the main menu, give the new filter a name, and specify the desired filtering rule. For example, to view all opened and closed tasks, specify the *status:closed status:open* query. Press *Save*, and you can access the created filter using the dedicated menu in the Filter bar.

More about Tasks

Tasks in Kanboard are not just simple entries, but containers that can hold a lot of useful information. And the application provides a dedicated interface for working with tasks. To switch to it, click on the desired task in any of Kanboard's views. This opens the task view, where you can see the task's summary, view the task's activity and its transitions (the task's movements between project columns) and analytics. The *Edit the task* command opens the familiar task editing dialog where you can
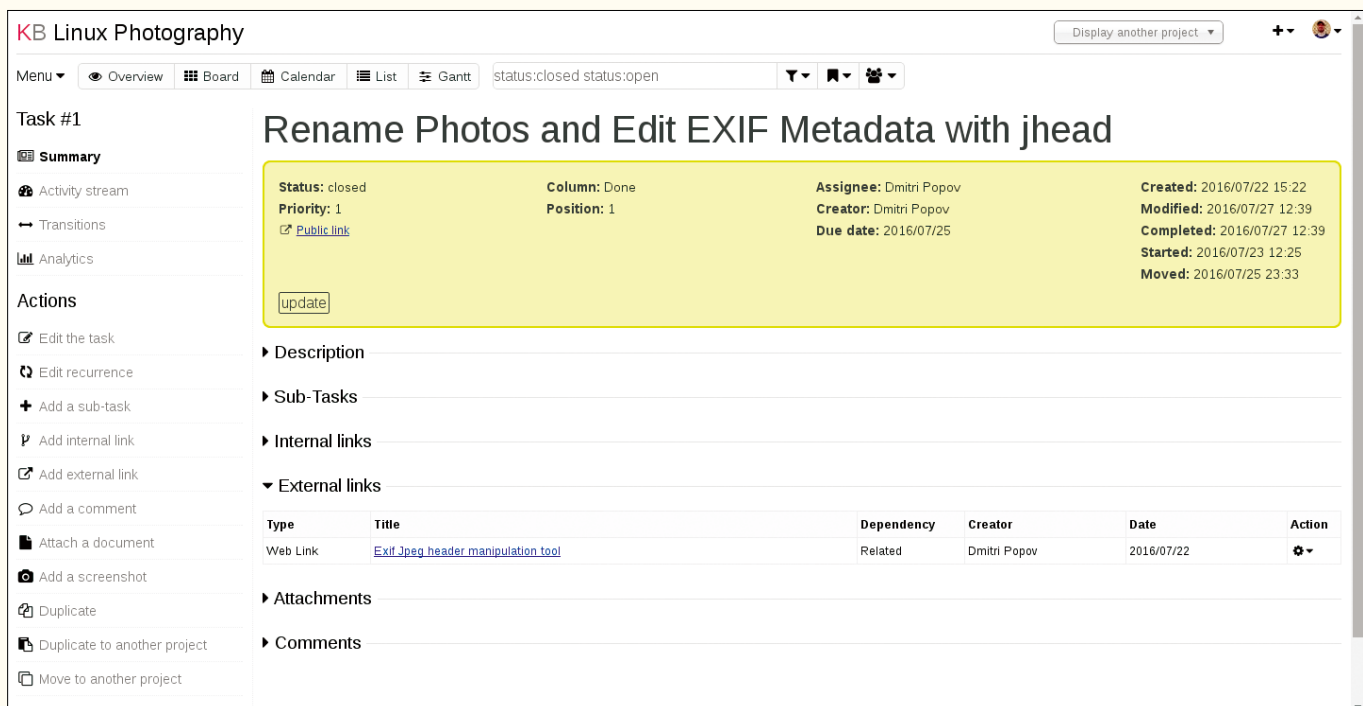
Figure 9: Editing a task.

modify the task's settings (Figure 9). Each task can contain a number of sub-tasks that can be added using the *Add a sub-task* command. This functionality can come in useful when you need to divide a complex task into smaller pieces. All sub-tasks appear in the *Sub-Tasks* section in the task view, and you can use the timer feature to track time spent on the particular sub-task.

Complex projects often include interdependent tasks. For example, you may have a project in which you can't start working on task Y before you complete task X, and any changes in the schedule for task X affect task Y and all other related tasks. Kanboard makes it possible to keep tabs on related tasks by creating internal links (i.e., relationships) between them. To specify an internal link for the current task, use the *Add internal link* command in the task view. In the *Add a new link* dialog, select the desired relationship from the *Label* drop-down list, and specify the target task in the *Task* field. Press *Save* to create the relationship. You can then use the *Gantt* view to manage connected tasks.

It's also possible to add external links to a task. This functionality can be useful for storing links that point to local files, websites, and attachments. Speaking of attachments, you can attach files to tasks, too. The *Add a screenshot* command lets you quickly attach screenshots to a task. Finally, the *Add a comment* command can be used to append comments that can be used for keeping task-related notes and discussions.

Final Word

Despite its apparent simplicity, Kanboard is a capable project management application loaded with useful features and tools. I covered only a tiny fraction of Kanboard's capabilities, and there is much more to this application than meets the eye. Fortunately, Kanboard comes with excellent documentation that can be accessed via the *Documentation* command in the User menu. Make sure to take a look at it if you want to get the most out of this excellent application. ■■■

INFO

[1] What is Kanban? *kanbanblog.com/explained*

[2] Kanboard: *kanboard.net*

[3] FreeOTP: *fedorahosted.org/freeotp*

Out of Bounds: Eye-catching photos made easy

Enhanced Drama

Including out-of-bounds effects in slide shows and presentations is bound to get the undivided attention of the audience. Gimp has simple tools to create these image effects. *By Karsten Günther*

O ut-of-bounds (OOB) photos refer to photos post-processed so that the contents fall outside of the borders, seemingly in three-dimensional (3D) perspective. These types of photos attract a lot of attention, make a visual theme much more interesting, and often accentuate a person or an object in the photograph. With some know-how, you can create OOB photos using only open source tools. For the most part, Gimp and its filters will suffice; however, some special-purpose photos will require Inkscape.

Typically, OOB images comprise three components: the original photograph; an image border, frequently with a distorted perspective; and a coordinated background. Simulating or adding emphasis to 3D aspects is icing on the cake. The 3D qualities are achieved with various types of shadows and other effects.

An Embedded Image

Photographs suitable for this type of montage are not what you necessarily imagine them to be. For example, a half-way-exposed photo that has captured a single object is often all you need. The entire object does not even have to be captured in the photo.

The trick is to select an area in the original photograph in such a way that it contains various constituent parts of the image. Later, these constituent parts will be situated in such a way that they conceal segments of a new border that you add. Gaining experience by tinkering helps you learn how to select the best area to use, and you will find that it really does help simply to experiment on your own.

Photographs that contain very prominent parts – or at least free-standing objects – are especially well suited to the OOB technique. Figure 1 shows the photos used in this article. Frequently, the embedded photos display a strong contrast ratio, or they have been post-processed to become brighter (Figure 2), lending a distinctive boldness. The opposite also delivers results worth viewing.

One technique is always present in the creation of OOB images: cropping. Usually this term refers to cutting photos to a particular size; however, the actual meaning of the word refers to separating a part of an image from the surrounding area. Various methods can be employed to crop an object in photographs. In the following sections, I demonstrate how to crop in Gimp.

Additional Layers

When cropping, always work on a copy of a layer so the original is preserved in the event that problems arise. To begin, you select the relevant area of the photo with the selection tool and copy it as a separate layer into the image.

As usual, Ctrl+C copies the selection to the clipboard and Ctrl+V pastes it to the image. Clicking *Layer | To New Layer* completes the action. You should make

◼ NIFTY SHADOW PLACEMENT

Gimp offers a very special function in its *Filters | Light and Shadow | Perspective* filter. The shadows it generates start at the object and extend according to the settings in the Perspective dialog. Shadows will look very different from drop shadows unless you use a steep angle between the light source and the object.

One drawback with this filter is that it does not have a preview because it has only been implemented as a Python script. It is difficult to guess what a parameter does by its name. *Angle* defines the direction of the shadow measured in the horizontal plane. Angles with fewer than 90 degrees create shadows to the right of the object. Those with greater than 90 degrees create shadows to the left.

The *Relative distance of horizon* parameter specifies the distance to an imaginary horizon. The documentation defines the relative distance as the distance from the baseline of the selection or layer. The height of the selection or layer counts as the unit of distance. The *Relative length of shadow* should not be longer than the *Relative distance of horizon*.

Large values for the length of the shadow can cause significant distortions. The *Blur radius* setting influences the edges of the shadow, and the *Color* setting controls the color of the shadow, whereas *Opacity* lets you determine how dark the areas of shadow appear. *Interpolation*, which is used to influence the perspective distortion of the shadow, has limited effect. Sometimes *None* leads to artifacts, although the Plugin will work very fast. The *Linear* option offers a good compromise between speed and quality.

The *Allow resizing* checkbox always calculates shadows completely, even if the shadows extend outside the selection and requires increasing the size of the of the shadow layer – and therefore possibly also the image size.

sure that the layer that has been generated contains all relevant parts of the image. In Gimp, this is how a transparent layer is created for use in subsequent work steps. In other programs, it may be necessary to add an alpha channel.

Now it is time to use the eraser tool to remove the areas surrounding the object. This tool lets you create a cropped area quickly. Usually you remove content, leaving the parts of the image you want intact. However, if you remove too much material from the pasted image in the transparent layer, you can recover it by holding the Alt key while "unerasing."

In many cases, it is not necessary to work out the entire design to the last detail. Precision becomes important only in places where the object will later touch the image border. At that point, even small mistakes can be noticeable.

Another area you will need to crop includes the places you will need later for perspective shadows. (See the box titled "Nifty Shadow Placement.") Normally, rough cropping suffices for this purpose. Gimp always calculates shadows at the transition between transparency and opaque areas in a layer and then blurs the details. In this way, small defects are obscured.

At this point, you should have two layers in the image, the original *background* and at least one partially cropped copy of the original, which should lie exactly on top of the other.
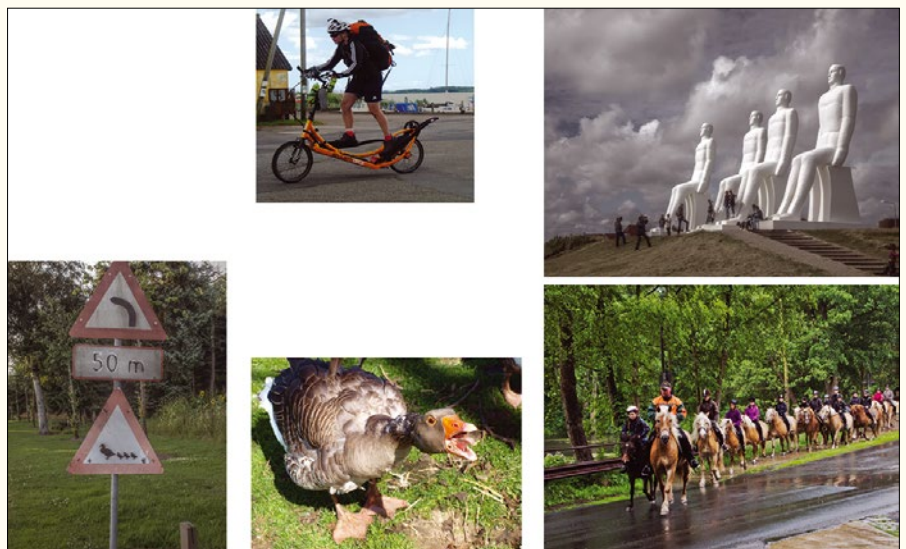

Figure 1: The original photographs used in this article.
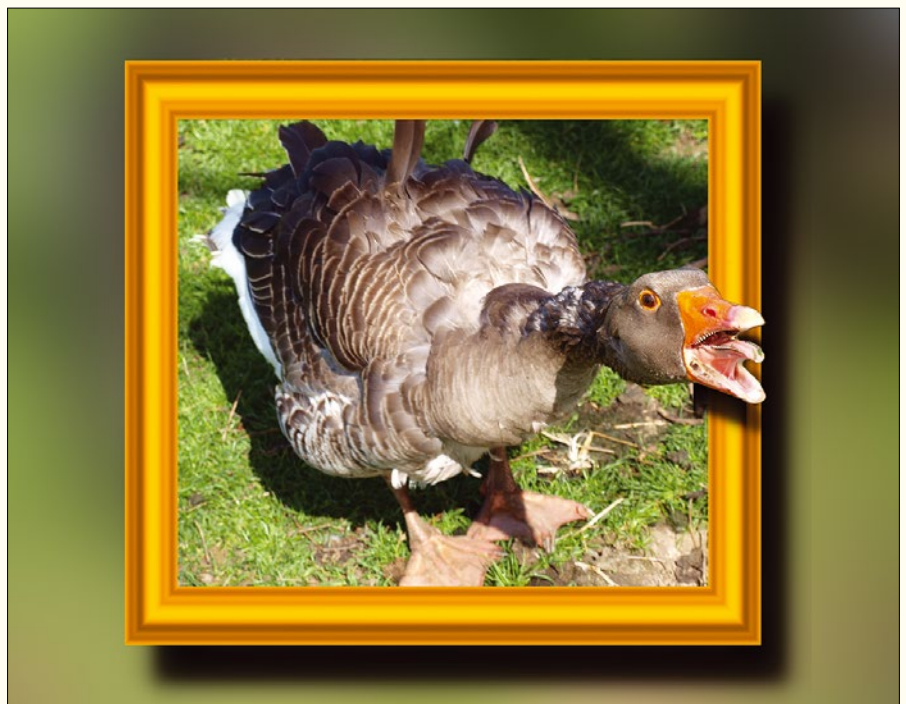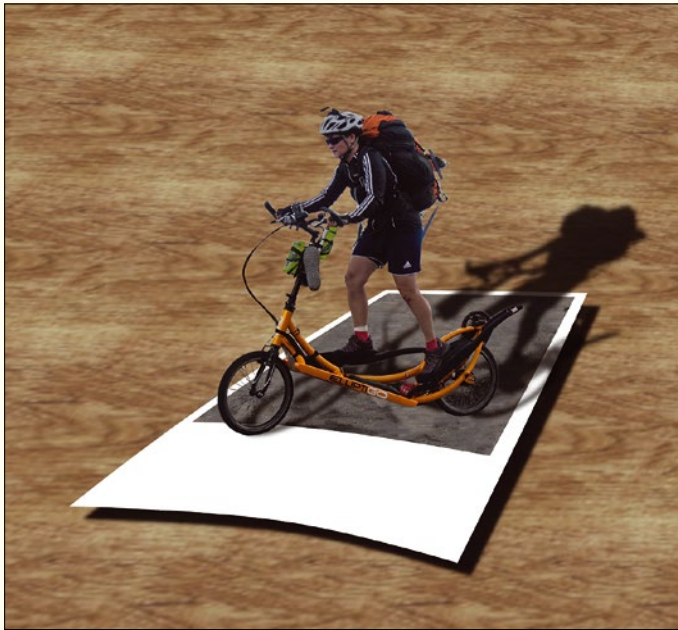

Figure 2: In this photograph, the tooth-like serrations of the beak have been brightened to accentuate the dramatic effect.

Figure 3: The frame for this image uses a somewhat curved Polaroid-type border and two shadow variations on a synthetic background.

Building Borders

Frame styles range from very simple to very complex. Because the frame separates the content from the background of the image, it should either look very discreet or be very obvious. In any event, it should fit the subject matter.

The simplest frames are created with lines you either draw or cut out. Gimp offers numerous possibilities for creating more complex frames. (See the box titled "Realistic Frames.") The easiest solution is to simulate a frame like those used for photos printed on paper by adding a white line that takes up a small bit of the image edges. Alternatively, you can use a Polaroid variation, which features a much wider area at the lower edge (Figure 3) of the photograph. To maintain the quality of the original, you always build the frame around it, thus increasing the size of the image.

Frames in OOB photos only connect to parts of the image, making them easy to create. For a simple frame, first draw a frame on the transparent layer that corresponds in size and shape to the final frame.

To transform the selection to a frame, use the *Select | Border* or *Select | Shrink* function. The first method creates rounded corners for the frame, whereas the second method does not. Next, you fill in the selection and reduce it in size to the width of the frame. Afterward, the selection, which is still active, should be deleted. The results will look like Figure 4.

Blurred Perception

The next step involves blurring the frame, which is somewhat tricky to do, even though it might not seem so at first glance. The ultimate effect is largely dependent on the settings chosen. You should be very precise in inserting the frame in a position that logically fits with the image.

Frequently, it is helpful to set up guides and paths around which the frame can then be created. To create a guide, first make sure that *View | Snap to Guides* is activated, then use the mouse to click and drag from one of the rulers that sit at the edge of the display. The use of guides ensures that the edges of the frame are parallel and perpendicular.

An alternative is to use *Tools | Paths* to create guides. Paths allow for random angles, and they can be traced with a paint tool and used with vanishing points (Figure 5), which should lie outside of the image surface to produce realistic effects. When *View | Snap to Active Path* is enabled, the reference points for a selection or another path are snapped to the vanishing line.

Figure 4: You can create a frame in a couple of different ways.

◼ REALISTIC FRAMES

To create realistic frames, you should put a shadow on the inner side of the frame to simulate thickness. A drop shadow outside the frame marks the distance of the frame to the background. An organized interface is helpful during this process, especially when you intentionally insert errors and blurring. Alternatively, you can use the *Bevel and Emboss* function from the Layer Effects tools [1].

The perspective function lets you simulate the location of the frame in the space. If you do not have a particular frame in mind but instead want it to look like a printed photograph, then you should make sure it does not appear too flat. Adding *Filters | Light and Shadow | Lighting Effects* highlights the sculptural impression.

Gimp itself offers limited possibilities for creating realistic frames. Therefore, it is best to switch over to G'MIC [2] or use the vector graphics program Inkscape [3], which is also well suited for constructing 3D frames. Nevertheless, image manipulation is always the first choice when the work involves the preparation and combination of layers for the entire composition.

Another option is to take a picture of a frame and use it for the OOB design. Note that the frame should be photographed head on from the front, or the top and the middle of the frame should serve as the central orientation point. This prevents unwanted distortions that would otherwise require more work later to correct. The plugin *Filters | Enhance | GimpLensfun* and *Filters | Distorts | Lens Distortion* can be helpful with this task.

You should avoid shadows that are not to be present in the final composition. Shadows on the frame need to agree with those in the original image. Glass frames should be avoided because they almost always create reflections. Brightness and color temperature should match as closely as possible to the original image.

No general rules determine how to adjust the perspective for the frame. The best plan is to create a frame and then experiment with several copies. When dealing with flat frames that are lying down or standing almost perpendicular, you should leave two edges parallel and then adjust the perspective on just one edge. Occasionally, randomly placed frames work well in the image you're working with, and interesting effects can be created by placing multiple frames in an image (Figure 6).

These methods let you build frames quickly and easily. The printed photos show that these simple frames can be improved relatively easily by creating a monochromatic frame on a separate layer.
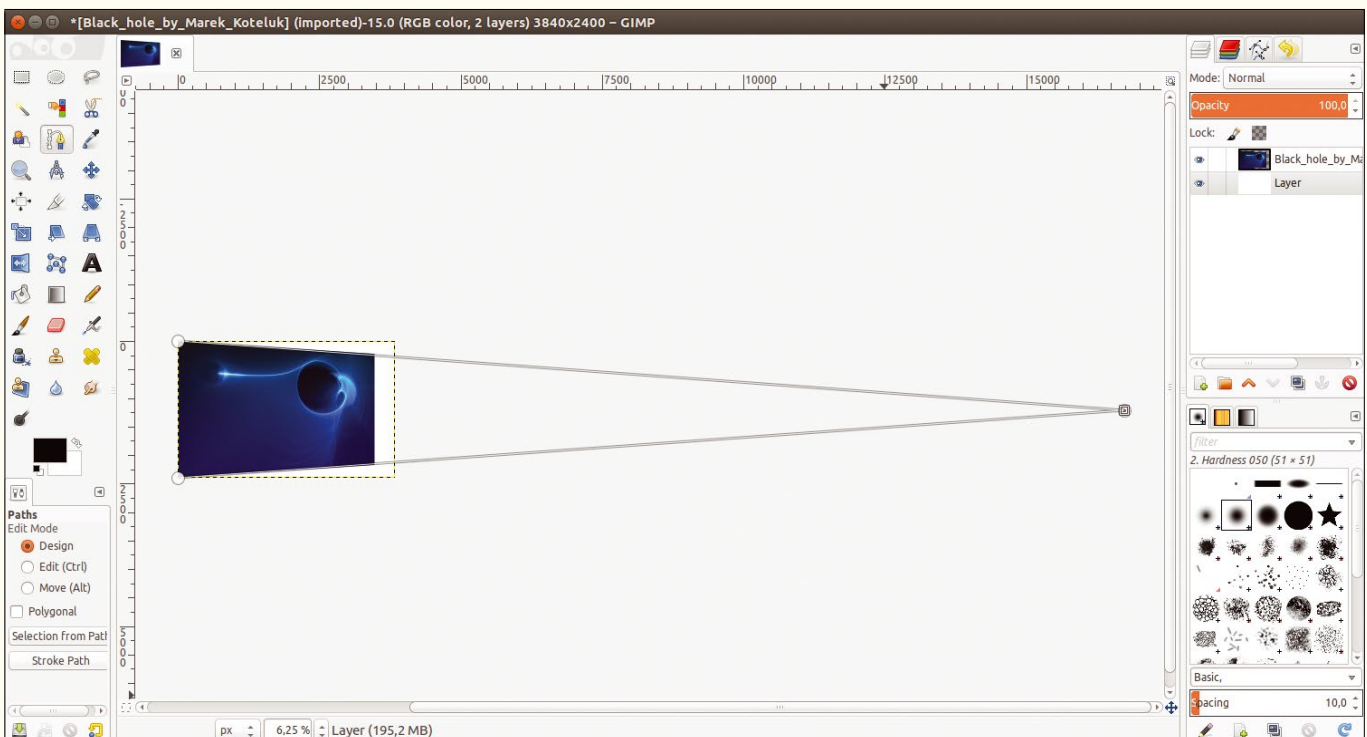

Figure 5: To work with vanishing points outside of the image, scale the image view and apply the path tool.

The frame can be curved with the *Filters | Distorts | Curve Bend* option. You should manipulate all of the layers equally by linking them together. Then you should drop shadows from the frame with *Filters | Light and Shadow | Drop Shadow*.

For a good alternative self-built frames, G'MIC has filters that automatically create frames. The global filter *Frame [painting]* is found in the *Frames* section (Figure 7). Although it has been around since 2012, the filter still seems largely unknown. With relatively few parameters, you can achieve a number of results.

The *Size (%)* parameter controls the thickness of the frame relative to the width of the image. *Contrast* and *Smoothness* specify the shape. A high value for *Smoothness* creates rounded shapes, and high contrast produces relatively thick frames. Clicking on the *Color* box lets you choose the basic color for the frame, and it comes with a pipette, so you can incorporate a color that occurs in the image.

The *Vignette* options can darken the corners of the image contained within the frame. Although usually an unwanted effect, it can create a realistic impression with relatively thick frames. The *Defects* options let you simulate small, random flaws in the lacquer of the frame in an almost photorealistic manner.

The *Serial number* parameter is almost completely undocumented. The only information about it comes from experiments (Figure 8). If you are working with a randomly shaped selection when calling this filter, G'MIC creates a frame for the bounding box; otherwise, it surrounds the entire current layer with a frame.

You can achieve a 3D effect with the OOB image by placing the frame layer over the original image layer, erasing the places where elements are supposed to be in front of the frame. You use the same tools to adjust the shape to the gaps in the frame. Alternatively, you could copy the layer with the original image and place it over the frame layer. In either case, you can use drop shadows to enhance the 3D effect of the frame and perspective shadows for the content (Figure 9).

Backgrounds

The backgrounds used in the image take on more meaning than you might otherwise imagine, because they surround and round out the image that has been created. Once again, you have basically two possibilities.

Plain backgrounds tend to emphasize an image rather than detract from it.

Figure 6: Multiple frames and partial color saturation create interesting effects.
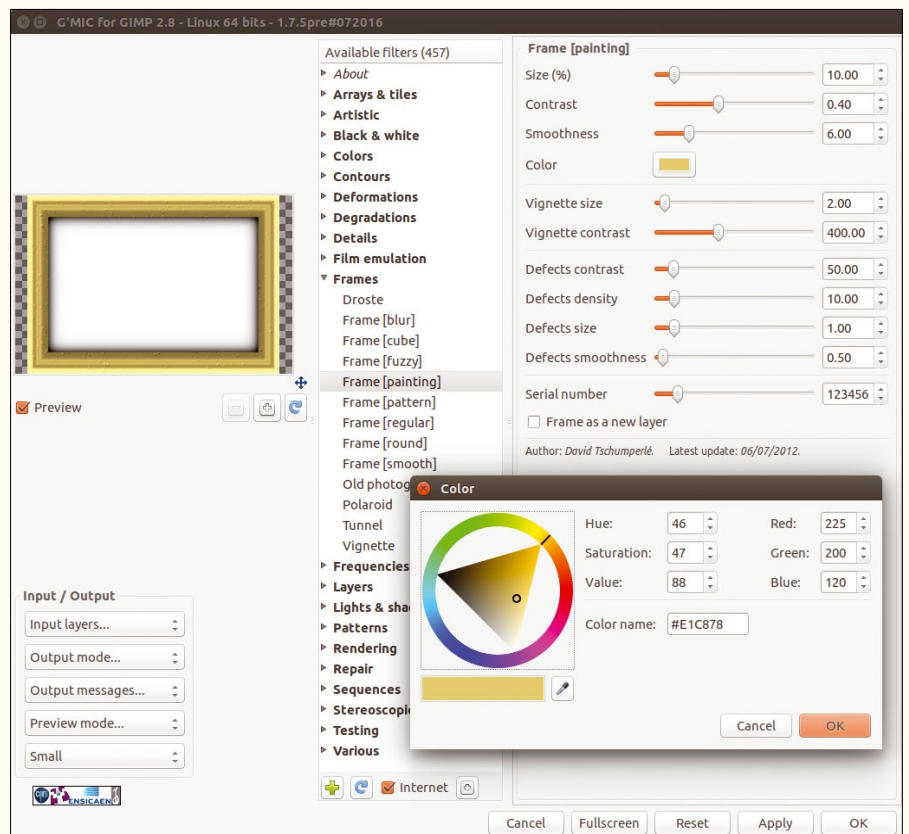
Figure 7: G'MIC offers an all-purpose filter for creating frames that makes it easy to achieve appealing results.

Figure 8: The *Serial number* effect controls the shape of the frame; however, it does so in a completely mysterious manner. The three versions show frames with minimal, mid, and maximum values.

This choice is suitable, for example, for flowers or other simple objects. One example is to use backgrounds extracted from the original image that you either blur or desaturate (Figure 6).

Another alternative uses complex backgrounds that then have to be incorporated into the entire composition. In rare cases, it is a good idea to forgo special backgrounds completely (Figure 10).

Most of the time it makes sense to use the Blend Tool to create a plain background. Various colors and shapes are available that let you blend from the color of the foreground to the color of the background. Both colors should come from the original image.

The background in Figure 8 starts with the blurred original image created with the G'MIC filter *Frame [blur]*. It was first applied to a layer in the background; then, I attached the original image on top. This process often leads to softer and more harmonious backgrounds than when the Gimp *Gaussian Blur* filter is used.

Conclusion

Out-of-bounds photos are created with the use of visually appealing tricks that require a significant increase in knowledge about effects in Gimp. To create these photos in a half-way convincing manner, you use numerous basic image manipulation functions and diverse elementary techniques. The adage "practice makes perfect" applies, but your reward will be the appeal of the resulting images to the viewer. ∎∎∎

INFO

[1] "Layer Effects" by JonStipe: *http://registry.gimp.org/node/186*

[2] G'MIC: *http://gmic.eu*

[3] Inkscape: *http://www.inkscape.org*

Figure 9: The combination of perspective shadows and drop shadows creates the impression of three dimensions.

Figure 10: In some cases, you do not need to use a special background.
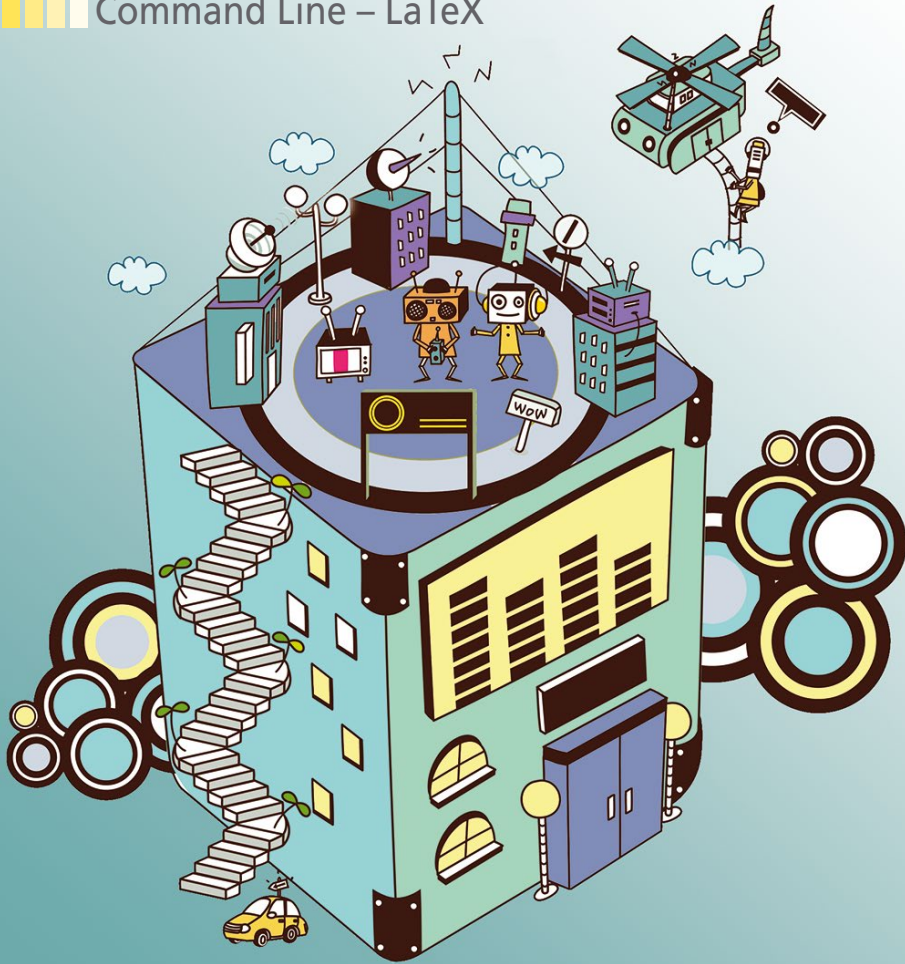
Tips and tools for managing LaTeX

Powerful Format

LaTeX has a reputation for being difficult to use, but mainly it just requires patience. We provide a quick overview to get you up and running. *By Bruce Byfield*

BRUCE BYFIELD

Bruce Byfield is a computer journalist and a freelance writer and editor specializing in free and open source software. Bruce's most recent book, *Designing with LibreOffice*, was released under a Creative Commons License in March 2016. You can buy or download his book at *http://designingwithlibreoffice.com/ download-buy/*. In addition to his writing projects, he also teaches live and e-learning courses. In his spare time, Bruce writes about Northwest Coast art. You can read more of his work at *http://brucebyfield.wordpress.com*.

L amport TeXt (LaTeX) is one of free software's legendary applications [1]. Before LibreOffice or AbiWord, it was the most advanced tool on Linux for formatting text. Even today, it is widely used in the academic sciences and in some publishing houses. Users speak of compiling their first lengthy document in LaTeX in the same way others talk of compiling their first Linux kernel, and several professors tell me that students regularly have to be prevented from

obsessing over LaTeX at the expense of their research.

LaTeX has a reputation for being difficult, but not because of its complexity. Instead, the difficulty lies in the size of the project. Adding to the difficulty is the number of editors designed for it, including extensions for both Vim and Emacs [2], as well as LyX, a graphical interface [3]. The difficulty in learning LaTeX does not lie in the concepts, as much as in finding the right tools.

At its heart, though, LaTeX is actually simple. Basically, it is a combination of a markup language similar to HTML and a collection of formatting macros. Creating a document in LaTeX consists of three steps:

1. Using Tex or some other editor to write the markup.
2. Compiling the LaTeX document.
3. Exporting to a usable file format with another script

Despite the many tools available for working with LaTeX, all these steps can be done using applications that are part of the default installation for most distributions. What follows is a quick overview to get you up and running.

Basic Structure of a LaTeX Document

To create a LaTeX document, you begin by opening any text editor. Figure 1 shows the structure of a simple TeX document. The first thing to notice is that the percentage sign (%) at the start of a line indicates a comment that is not printed. A backslash (\) at the start indicates a piece of markup. Often, the markup is a reference to a formatting macro and which formatting choice(s) are being used, with the macro coming first and the choice following in curly braces. For instance, \documentclass{article} defines the type of document as an article.

The next thing to notice is that the document is divided into sections, with the beginning and end of each section

```
% first.text - minimal document
\documentclass{article}
\begin{document}
This is a basic sample.
\end{document}
```

Figure 1: All TeX documents have the same minimal structure.

```
% first.text - minimal document

\documentclass[12pt,letterpaper,oneside,draft]{article}

\usepackage{graphicx}
\graphicspath { {screenshots/} }

\title{Basic LaTeX}
\author{Bruce Byfield}
\date{August 2016}
```

Figure 2: A typical preamble in a document.

clearly marked. Everything above `\begin{document}` is known as the preamble or the top matter.

The preamble is the place for meta-information – not only introductory comments, but also information like the author, title, and the date, such as `\author{Bruce Byfield}`. The preamble also includes high-level formatting for the entire document (except where overridden by a section), including the default font size and the paper size, such as `\documentclass[12pt,letterpaper,oneside,draft]{report}`, the formatting being added in square brackets after the macro. For convenience, you can divide this information into stanzas with a space between each one. The average preamble then would probably look more like Figure 2.

You can start adding content below `\begin{document}`. It can be further subdivided by other section types. For the most part, the content is simply typed, with a blank line indicating a new paragraph. None of the content, of course, will be formatted until you compile. And, if you actually want to print a backslash, you must add another backslash (\\) so that it is not interpreted as markup. That covers most of the mechanics of writing markup that you need to know.

Basic Formatting

The rest of using LaTeX is almost entirely looking up which macros you need to get the results you want. Fortunately, LaTeX is exhaustively documented and, with some patience, you should be able to find the information you need. All the same, when drafting, you should compile frequently and be prepared, if necessary, to continually tweak your markup. If you do the same type of document repeatedly, save satisfactory results as templates so you don't need to test more than once.

You can usually save time by defining a document type and its formatting options (see Table 1) in the preamble and letting the macro do most of the work for you. For general purposes, use article as the document type. Other common classes include book, letter, report (including thesis), and beamer (presentations).

For more control over page appearance, you can also add some non-standard document classes [4] by hunting down their packages on the Internet and placing those

TABLE 1: Common Options for Most Document Classes

Formatting	Markup	Comments
Paper size	\usepackage[a4paper]{geometry}	Paper sizes include letterpaper & legalpaper
Orientation	\usepackage[landscape]{geometry}	Values are landscape & portrait
Margins	\usepackage[top=length, bottom=length, left=length, right=length]{geometry}	Length is in centimeters or inches
Headers and footers	\pagestyle{style}	Applies to current and subsequent pages. Choices are empty (no header or footer), plain (foot includes page number), or headings (header contains information defined by document class)
Headers and footers	\this pagestyle{style}	Same as \pagestyle, but only for current page

packages in the same directory as the Tex file. If some sections define their own pages, you can override them by placing `\pagestyle{empty}` in the preamble and defining the `pagestyle` as need throughout the document.

Another way to reduce the amount of formatting is to define a section, such as `\chapter{''TITLE''}` or `\section{''TITLE''}`. But, using sections can become complicated, especially if you want to repress the numbering that many sections automatically add.

For the equivalent of word processor character styles, you can use `\textit` for italics, `\textbf` for bold, and `\underline` for underlining. In each case, follow the basic command with the text to format in curly braces. In other words, to place "equivalent" in the first sentence of this paragraph in italics, you would enter:

```
For the \textit{equivalent} of word processor character styles
```

Ordered lists start with `\begin{enumerate}` and end with `\end{enumerate}`, whereas unordered lists are marked by `\begin{itemize}` and `\end{itemize}`. Both types of list can have as many entries as required, but each item must begin on a separate line that begins with `\item`.

To use graphics, place `\usepackage{graphicx}` and `\graphicspath/{ {FOLDER} }` in the preamble, being careful to end the graphics path with a forward slash. At the position where you want to place the graphic, add `\include[FILE]`.

Modern LaTex supports both EPS and PNG files, but you do not need to include any extension when adding the image to the document. You can change the size of the graphic with `\includegraphics[scale = RaTIO]`, where 1.0 is the original size or change the dimensions with `\includegraphics[width=SIZE height=SIZE]`.

These are far from the only formatting options you are likely to want to use with LaTeX, and they do not address challenges such as positioning a page break where you want it. However, they should be enough for many documents.

Compiling and Outputting

When you are finished creating the file, save it and then run the command `latex FILE`. The result is a file of the same name, except with a DVI extension. A logfile of the same name is also created, allowing you help with troubleshooting the markup. You can also find various viewers, such as advi, to inspect the file visually.

If there are no errors, you can use the DVI file to export to four formats: postscript (PS), PDF, RTF, and HTML. There are many scripts for these exports, but div2ps, ps2pdf, latext2rtf, and latext2html should do for most purposes.

Although many instructions warn of a possible loss of formatting when exporting to RTF or HTML, I have personally never encountered any. If you do, however, a postscript or PDF export should be more reliable.

Learning LaTeX

A full description of LaTeX literally takes a book. But, once you understand how it works, designing a LaTeX document is mainly a matter of patience. Dozens of websites are designed to educate new users, and, although some are outdated in various details, the basic structure has not changed over the years. LaTeX's consistency is strong enough that, with the examples given here, you should quickly understand what you are doing.

While you are learning LaTeX, it may be helpful to start each document by noting the formatting you will need and then bookmarking useful sites. Additional features, such as captions, tables, and bibliographies should be structurally consistent to those given, differing only in the resources summoned by the file [5].

If LaTex has a fault, it is that only an experienced user can hope to customize it completely. However, in most cases, its packages do most of the work for you. Almost always, the result is a professional, if unvaried design. Aficionados sometimes over-estimate LaTeX's ability to customize compared to LibreOffice Writer, but, in experienced hands, it is definitely more versatile than Microsoft Word or Google Docs. ◼◼◼

◼INFO

[1] LaTeX: *http://www.latex-project.org/*

[2] TeX editors: *https://en.wikipedia.org/wiki/Comparison_of_TeX_editors*

[3] LyX: *http://www.lyx.org*

[4] Document classes: *https://en.wikibooks.org/wiki/LaTeX/Document_Structure#Document_classes*

[5] Getting to grips with LaTeX: *http://www.andy-roberts.net/writing/latex/page_layout*

Zack's Kernel News

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

By Zack Brown

ISH Support

Srinivas Pandruvada from Intel announced a set of patches to better support integrated sensor hub (ISH) chips under Linux. ISH chips support motion detection and other sensors, more tightly integrated than earlier external sensor hardware. Srinivas's code didn't change the application binary interface (ABI) at all, so existing software should run fine and properly detect and use ISH chips without needing to be recompiled. But this initial release was mostly intended as a request for comments, rather than an actual submission for inclusion in the kernel tree.

Atri Bhattacharya, Daniel Turton, and Grant Likely all thanked Srinivas for sending these patches along. Atri reported that the patches correctly detected the ISH chip on his Lenovo Yoga 260, though it had trouble using the ISH sensors. In particular, suspend and wakeup became very slow, and he noticed a lot of time outs in the logs.

Daniel, also using a Yoga 260, reported that the kernel compiled OK with Srinivas's patches but wouldn't detect the ISH chip. He posted some debugging information for Srinivas.

Grant, testing on an HP Spectre x360 G2, successfully built the kernel, which successfully detected the ISH chip, and successfully created control files under `/sys/bus/iio/devices/`, but he saw no obvious way to test the code.

To Grant, Srinivas offered a few suggestions: run `cat` on the files; modify them to control screen rotation and brightness. Grant tried some of these things and reported back that the system did detect the ISH chip accelerometers, but the sensor events didn't seem to be reaching the kernel.

Srinivas offered suggestions to fix that issue, as well as Atri's issue with suspend and wakeup. There was a bunch more debugging, also joined by Bastien Nocera. No solutions were found during the course of discussion.

I really enjoy seeing companies like Intel submitting code early and concerning themselves more with collaborating with the kernel developers than presenting a perfectly working set of patches. It took a long time for some of those big industry players to get used to a public development process, in an era where marketing and legal departments are terrified of anything going even slightly wrong.

Per-Task Livepatching

Josh Poimboeuf posted some patches against livepatch – the kernel feature that lets users update a running kernel without having to reboot. Specifically, Josh had implemented per-task updates, where the kernel would migrate running processes over to using a given patch one by one, as it deemed each task ready to make the switch. This is good for patches that alter the data that's available to a process, or that in some way change the internal kernel structures that need to be referenced.

Josh's code used several techniques to determine when a process was ready to receive a patch. Presumably these techniques will be the things that future generations of kernel hackers will debate and improve. But Josh also included a feature to apply patches immediately, if they were known to change no semantics or data.

Petr Mladek had some technique comments but overall averred, "It is really great work! I am checking this patch from left, right, top, and even bottom, and all seems to work well together."

Miroslav Benes and Jessica Yu also offered some technical criticisms. Jessica asked, "What about tasks sleeping on affected functions in uninterruptible sleep (possibly indefinitely)? Since all signals are ignored, we wouldn't be able to patch those tasks in this way, right? Would that be an unsupported case?"

Jiri Kosina agreed that this could be an issue but didn't see any solution beyond documenting it as a possibility, or perhaps trying to detect the situation and alert the user. And, Josh said that Jessica's case would probably be a very rare occurrence and "not something we need to worry about for now."

Related to Jiri's idea of a user alert, David Laight suggested, "please can we have a flag for the sleep and/or process so that an uninterruptible sleep doesn't trigger the 'hung task' detector and also stops the process counting towards the 'load average'. In particular some kernel threads are not signalable, and do not want to be woken by signals (they exit on a specific request)."

They continued to discuss the technical details briefly. Livepatching is one of those weird, incredible things that would have been deemed impossible 20 years ago, like Linus's

⬛ 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**.

famous pronouncement that Linux would probably only ever support the 386 processor. Somehow, these big, unlikely features are allowed to come gradually into the kernel, as long as they don't spread their insanity through the rest of the code.

TSN Support in the Works

Henrik Austad had been working on a Time Sensitive Network (TSN) driver for a while and finally decided to send it in for a first review. TSN is mostly used for streaming data across a network, either media or communications.

Henrik's work was met with a serious rebuke from Richard Cochran, who said:

"Your series does not even begin to address the real issues. I did not review the patches too carefully (because the important stuff is missing), but surely configfs is the wrong interface for this. In the end, we will be able to support TSN using the existing networking and audio interfaces, adding appropriate extensions.

Your patch features a buffer shared by networking and audio. This isn't strictly necessary for TSN, and it may be harmful. The Listeners are supposed to calculate the delay from frame reception to the DA conversion. They can easily include the time needed for a userspace program to parse the frames, copy (and combine/convert) the data, and restart the audio transfer. A flexible TSN implementation will leave all of the format and encoding task to the userland. After all, TSN will some include more that just AV data, as you know.

Let's take a look at the big picture. One aspect of TSN is already fully supported, namely the gPTP. Using the linuxptp user stack and a modern kernel, you have a complete 802.1AS-2011 solution.

Here is what is missing to support audio TSN:

- *User Space*
  1. *A proper userland stack for AVDECC, MAAP, FQTSS, and so on. The OpenAVB project does not offer much beyond simple examples.*
  2. *A userspace audio application that puts it all together, making use of the services in #1, the linuxptp gPTP service, the ALSA services, and the network connections. This program will have all the knowledge about packet formats, AV encodings, and the local HW capabilities. This program cannot yet be written, as we still need some kernel work in the audio and networking subsystems.*

- *Kernel Space*
  1. *Providing frames with a future transmit time. For normal sockets, this can be in the CMESG data. For mmap'ed buffers, we will need a new format. (I think Arnd is working on a new layout.)*
  2. *Time-based qdisc for transmitted frames. For MACs that support this (like the i210), we only have to place the frame into the correct queue. For normal HW, we want to be able to reserve a time window in which non-TSN frames are blocked. This is some work, but in the end it should be a generic solution that not only works "perfectly" with TSN HW but also provides best effort service using any NIC.*
  3. *ALSA support for tunable AD/DA clocks. The rate of the Listener's DA clock must match that of the Talker and the other Listeners. Either you adjust it in HW using a VCO or similar, or you do adaptive sample rate conversion in the application. (And, that is another reason for *not* having a shared kernel buffer.) For the Talker, either you adjust the AD clock to match the PTP time, or you measure the frequency offset.*
  4. *ALSA support for time triggered playback. The patch series completely ignore the critical issue of media clock recovery. The Listener must buffer the stream in order to play it exactly at a specified time. It cannot simply send the stream*

ASAP to the audio HW, because some other Listener might need longer. AFAICT, there is nothing in ALSA that allows you to say, sample X should be played at time Y. These are some ideas about implementing TSN. Maybe some of it is wrong (especially about ALSA), but we definitely need a proper design to get the kernel parts right. There is plenty of work to do, but we really don't need some hacky, in-kernel buffer with hard coded audio formats."

Arnd Bergmann replied to Richard's kernel item #1, regarding transit timings. Arnd said, "the timestamps in the current v3 format are sufficient until 2106 as long as we treat them as 'unsigned', so we don't need the new format for y2038, but if we get a new format, that should definitely use 64-bit timestamps."

Henrik also replied to Richard, saying that his patches were intended to get the discussion going, not to be a real solution to the various problems. So, he tried to address each of Richard's objections in turn. Takashi Sakamoto also joined in with his own technical critique of Henrik's patches. The discussion delved into specific standards documents and kernel behaviors.

Overall, it seems as though Henrik's patches got some of the right people talking; though it's clear that in terms of actual code and feature support, TSN remains in a very early stage of development – essentially nowhere. That is exactly what I found so cool about this discussion. It helped clarify everyone's understanding of what a proper set of features would look like, and who are likely to be the key reviewers of future patches.

Peter Zijlstra wanted to address an issue with spinlocks that had bitten a bunch of users. Specifically, he'd noticed cases where code waiting for a locked resource might check the lock earlier than expected, and miss out on crucial state changes. It was a highly internal-to-the-code issue, and Linus Torvalds pointed out the problems with conceiving of it that way:

"NAK.

We don't start adding more of this 'after_ctrl_dep' crap.

It's completely impossible to understand, and even people who have been locking experts have gotten it wrong.

So it is *completely* unacceptable to have it in drivers."

This needs to be either hidden inside the basic spinlock functions, _or_ it needs to be a clear and unambiguous interface. Anything that starts talking about control dependencies is not it.

Note that this really is about naming and use, not about implementation. So something like 'spin_sync_after_unlock_wait()' is acceptable, even if the actual _implementation_ were to be exactly the same as the 'after_ctrl_dep()' crap.

The difference is that one talks about incomprehensible implementation details that nobody outside of the person who *implemented* the spinlock code is supposed to understand (and, seriously, I have my doubts even the spinlock implementer understands it, judging by the last time this happened), and the other is a much simpler semantic guarantee.

So, don't talk about 'acquire'. And most certainly don't talk about 'control dependencies'. Not if we end up having things like *drivers* using this like in this example libata."

At this point, Tejun Heo came in, saying that Peter's issue could possibly be resolved by simply removing the offending code from the kernel, as it wasn't needed anymore. Peter replied, "that would be great; I was sorta lost in there, but it looked like if you need the spin_unlock_wait() you also need the extra barrier thing. If you can remove it, better still."

Meanwhile, Peter said he'd also be fine shoving his fix into the main spinlock implementation as per Linus's recommendation.

The discussion petered out around there. But, it's interesting because the kernel's locking features have gone through such hellish transformations over time, in particular with the transition from the Big Kernel Lock (BKL) to the various smaller locks now sprinkled throughout the kernel. Typically, kernel locks make the code more and more complicated even as they allow support for smoother and smoother multitasking. There's a strong pressure to reduce locking complexity and to make locks as intuitive as possible. Hence, Linus's uncompromising use of the hammer.

Closing Security Holes in User Namespaces

Serge E. Hallyn posted a patch to add some cool magic to file capabilities in namespaces. Linux uses namespaces to corral resources into virtual systems, essentially allowing multiple whole Linux operating systems to run on top of a single host. The problem was that a hostile user could create their own namespace, assign themselves the identity of a root user, and then use their higher privilege level to change the extended attributes (xattrs) of a file to give improper access to regular users.

With Serge's patch, whenever the root user of a given namespace attempted to alter security xattrs, the kernel would sneak past the root identity and rely on the credentials of the original user instead. If the original user lacked the necessary credentials, the kernel would use an alternative security xattr that would allow the process to complete its work, without actually giving the targeted files any security holes.
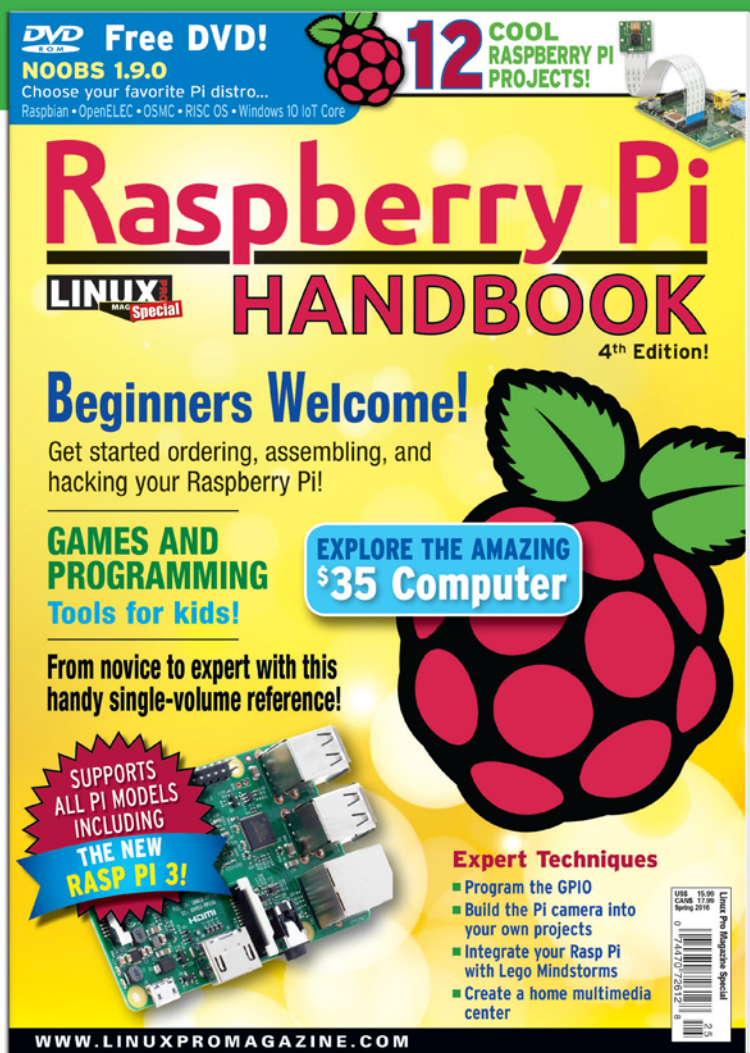
Mimi Zohar liked the whole sneaky process, especially the part where userspace didn't need to keep track of whether it was in or out of a namespace. That's one of the holy grails of virtualization – to make the virtual system seem identical to a standard instance of Linux running directly on its own hardware. But, because of various complex security considerations, it's very difficult to provide absolutely every feature of regular Linux in a virtualized environment. Serge's patch was an attempt to address one instance of that type of issue.

Meanwhile, Eric W. Biederman had some criticisms of Serge's whole concept. He didn't like that Serge's code would allow an equivalent, though less dangerous, xattr to be written instead of the more dangerous one desired by the hostile attacker. He felt that these two xattrs were essentially identical but ran the risk of getting out of sync. Eric preferred to just allow the user to write the xattr, but record the fact that it was the user doing it and not root. This way, he said, the security would not be compromised.

Mimi asked if Eric's suggestion would allow multiple instances of the same xattr, and Serge explained, "No, but we don't actually want that anyway. The current behavior for security capability is that it works in all user namespaces. So, we want to continue the behavior that if root in the init_user_ns sets a capability, that [it] works in all namespaces. Allowing other namespaces to set the capability would only be confusing." ∎∎∎

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*.

LinuxCon Europe

Date: October 4–6, 2016

Location: Berlin, Germany

Website: *http://events.linuxfoundation. org/events/linuxcon-europe*

Learn from the leading maintainers, developers, and project leads in the Linux community from around the world. Choose from among 100+ sessions, ranging from tutorials to deep technical dives and everything in between.

OpenStack Summit

Date: October 25–28, 2016

Location: Barcelona, Spain

Website: *https://www.openstack.org/ summit/*

Welcome to users, developers, and administrators of OpenStack cloud software. This five-day conference includes keynotes, breakout sessions, and working sessions. Learn about new products in the ecosystem and participate in hands-on workshops.

Node.js Interactive

Date: November 29–December 2, 2016

Location: Austin, Texas

Website: *http://events.linuxfoundation. org/events/node-interactive*

Node.js Interactive is the marquee event for JavaScript developers, companies that rely on Node.js, and vendors that support them with tools, training, and other services. Join the fun with Breakout, Code & Learn, and Un-conference sessions, as well as the

EVENTS

ownCloud Contributor Conference 2016	September 9–15	Berlin, Germany	https://conference.owncloud.org/conference/oCC2016/
Node Interactive Europe	September 15–18	Amsterdam, Netherlands	http://events.linuxfoundation.org/events/node-interactive-europe
Kieler Open Source and Linux-Tage	September 16–17	Kiel, Germany	http://www.kilux.de/
Open Source Backup Conference	September 26-27	Cologne, Germany	http://osbconf.org/
LPI Partner Day 2016	September 21–22	Kassel, Germany	https://www.lpice.eu/cfp
systemd.conf2016	September 28– October 1	Berlin, Germany	https://conf.systemd.io/
LinuxCon Europe	October 4–6	Berlin, Germany	http://events.linuxfoundation.org/events/linuxcon-europe
ContainerCon Europe	October 4–6	Berlin, Germany	http://events.linuxfoundation.org/events/containercon-europe
Icinga Camp San Diego 2016	October 18	San Diego, California	https://www.icinga.org/community/events/icinga-camp-san-diego/
Linux Presentation Day	October 22	Berlin, Germany	http://www.linux-presentation-day.org/
eclipsecon Europe	October 25-27	Ludwigsburg, Germany	https://www.eclipsecon.org/europe2016/
OpenStack Summit 2016	October 25–28	Barcelona, Spain	https://www.openstack.org/summit/
Node.js Interactive North America	November 29– December 2	Austin, Texas	http://events.linuxfoundation.org/events/node-interactive
Open Source Monitoring Conference	November 29– December 2	Nürnberg, Germany	https://www.netways.de/events_schulungen/osmc/uebersicht/
IT-Tage 2016	December 12–15	Frankfurt, Germany	http://www.ittage.informatik-aktuell.de/

Images © Alex White, 123RF.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

Erik Bärwaldt	60
Swapnil Bhartiya	8
Zack Brown	92
Bruce Byfield	88
Joe Casad	3
Sabine Drasnin	42
Karsten Günther	80
Dr. Harald Jele	28
Heike Jurzik	42
Jan Kleinert	24
Klaus Knopper	58
Charly Kühnast	66
Martin Loschwitz	18
Fabian Melters	14
Andreas Möller	48
Dmitri Popov	74
Mike Schilli	68
Tim Schürmann	24
Martin Steigerwald	24
Uwe Vollbracht	38

Issue 192 / November 2016

25 Years of Linux!

Next month we celebrate 25 beautiful years of the incredibly powerful and versatile Linux operating system. We'll ask some noted Linux personalities for their favorite memories, and we'll take a look at the colorful and sometimes controversial communication style of the Linux kernel community. This special anniversary issue also includes the latest edition of the rarely seen Linux Magazine Archive DVD, with our entire history – 191 complete issues of Linux Magazine – on a single searchable disc.

THE COMPLETE LINUX MAGAZINE ARCHIVE

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*