FREE DVD

4 COOL DESKTOP TOOLS REINVENT YOUR WORK LIFE

NOW FEATURING LINUXVOICE

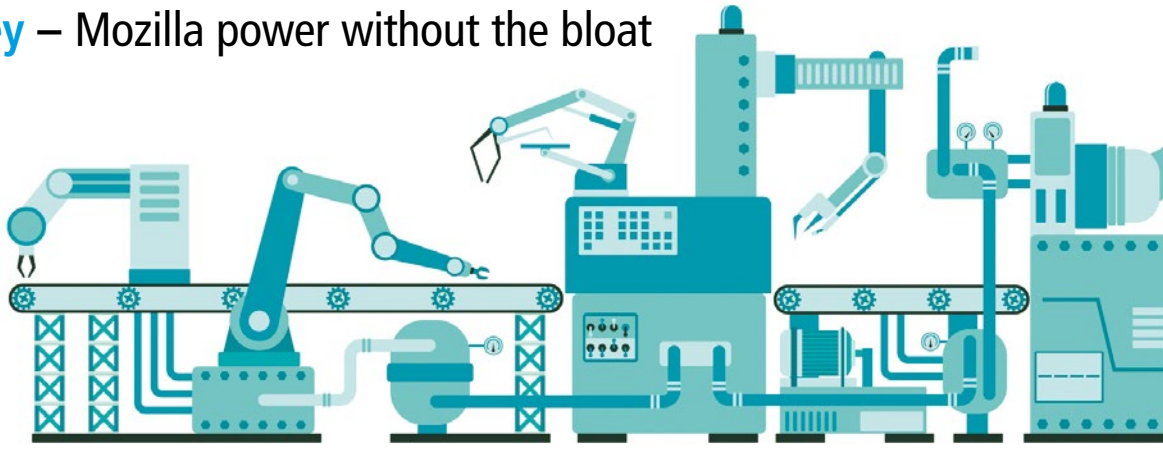
LINUX PRO MAGAZINE

MAY 2017

4 COOL DESKTOP TOOLS

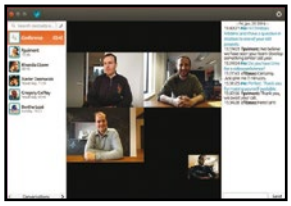
REINVENT YOUR WORK LIFE WITH:

- **Zim** – Manage your thoughts and notes with this handy personal wiki
- **OnlyOffice** – Office suite with local cloud option for easy collaboration
- **Getting Things Gnome** – Take control of your task list
- **SeaMonkey** – Mozilla power without the bloat



AryaLinux
Create your own Linux distribution

Endless OS
Linux for emerging countries



Ring
Secure text, voice, and video with a FOSS edge

SmartThings
Surviving a power outage in your smart home

Subgraph OS
Privacy distro with the emphasis on ease of use

LINUXVOICE

- Discover the world of IRC
- The Free vs Open dichotomy
- The future of routers

FOSSPicks
- Hardware automation with Open Stage Control
- Peruse comic book reader

Tutorial
Browse the web from the command line

Issue 198
May 2017
US$ 15.99
CAN$ 17.99

HETZNER ONLINE

LIGHTNING QUICK
RAZOR SHARP

Dedicated Root Server EX51-SSD-GPU

Intel® Core™ i7-6700
Quad-Core Skylake Processor
64 GB DDR4 RAM
2 x 500 GB SATA 6 Gb/s SSD
GeForce® GTX 1080 graphics card
100 GB Backup Space
50 TB traffic inclusive*
No minimum contract
Setup Fee $105

monthly $ **105**

The ideal solution for resource-intensive calculations.

The new dedicated root server EX51-SSD-GPU houses an incredibly powerful graphics card, the top-of-the-line GeForce® GTX 1080, which accelerates all sorts of graphics and video processing, such as 3D rendering.

www.hetzner.de/us

* There are no charges for overage. We will permanently restrict the connection speed if more than 50 TB/month are used. Optionally, the limit can be permanently cancelled by committing to pay $1.30 per additional TB used.

MAKE IT BETTER

Dear Reader,

Sir Tim Berners-Lee has something to say about the state of the World Wide Web. What does he know about the Web? Lots, because he invented it.

Berners-Lee was working as a fellow at the CERN lab in Geneva, Switzerland in 1989 when he noticed that, if the people working on hypertext and the people working on TCP/IP networking would get together, the result might be something very interesting. He wrote the first web browser, and the first web server, and he helped develop the specification for the magic string we know today as a URL. He also founded the World Wide Web Consortium and has been active in developing and popularizing the web ever since.

So when Tim Berners-Lee has something to say about the web, we would be wise to listen. Berners-Lee posted a message [1] at the webfoundation.org site to mark the 28th anniversary of the web. In the post, he wrote, "I'm becoming increasingly worried about three new trends, which I believe we must tackle in order for the web to fulfill its true potential as a tool which serves all of humanity."

The three trends that worry Tim Berners-Lee are:

- We've lost control of our data – the dominant business model for the web calls for free content in exchange for personal data, and we have very little control over how that data is used.
- It's too easy for misinformation to spread on the web – the pay-per-click model that search engines and social media sites use for compensating content providers emphasizes sensationalism over accuracy, meaning that "…misinformation, or 'fake news', which is surprising, shocking, or designed to appeal to our biases, can spread like wildfire."
- Political advertising online needs transparency and understanding – algorithm-based ad services can manipulate the information provided for individual voters in complex ways, warping the context for electoral decisions.

Although no solutions for these vast and intractable problems appear on the horizon, the first two issues have received significant coverage in the press – including plenty of attention in this column you are now reading. The third issue has received comparatively less attention, and Tim Berners-Lee deserves credit for adding the important topic of online political advertising to the conversation.

According to Berners-Lee, "The fact that most people get their information from just a few platforms and the increasing sophistication of algorithms drawing upon rich pools of personal data, means that political campaigns are now building individual adverts targeted directly at users. One source [2] suggests that in the 2016 US election, as many as 50,000 variations of adverts were being served every single day on Facebook, a near-impossible situation to monitor. And there are suggestions that some political adverts – in the US and around the world – are being used in unethical ways – to point voters to fake news sites, for instance, or to keep others away from the polls."

According to the post, the Web Foundation is working on a five-year strategy [3] to address these issues by "researching the problems in more detail, coming up with proactive policy solutions, and bringing together coalitions to drive progress towards a web that gives equal power and opportunity to all." The group plans to explore a variety of options, including alternative revenue models, data pods for protecting personal information, and mechanisms for encouraging gate keepers such as Google and Facebook to combat misinformation.

The scope and depth of these problems makes it difficult to believe that one think tank of open web advocates will be able to solve them all alone, but the real work begins with churning up new ideas and starting to talk about them. Thanks to Tim Berners-Lee and the Web Foundation for daring to dream of a better, healthier web.

Joe Casad,
Editor in Chief

INFO

[1] "Three Challenges for the Web, According to It's Inventor": *http://webfoundation.org/2017/03/web-turns-28-letter/*

[2] "Google, Democracy, and the Truth about Internet Search": *https://www.theguardian.com/technology/2016/dec/04/google-democracy-truth-internet-search-facebook*

[3] "Delivering Digital Equality: The Web Foundation's 2017-2022 Strategy": *http://webfoundation.org/2017/02/delivering-digital-equality-the-web-foundations-2017-2022-strategy/*

LINUX MAGAZINE

WHAT'S INSIDE

If you're like many Linux users, that perfect desktop tool you really need does not always appear in your default configuration. The Linux repositories are a treasure trove of interesting and practical applications, and we highlight four in this issue: the Zim personal wiki, OnlyOffice, Getting Things Gnome, and Mozilla's SeaMonkey Internet suite.

Also inside:

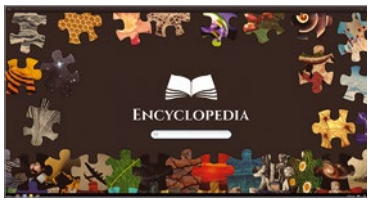
- **Subgraph** – a privacy-conscious OS that doesn't require an expert (page 42).
- **AryaLinux** – A Linux distro designed for users who want to roll their own system from source code (page 54).

You'll find lots more in LinuxVoice, including a look at the Linux Embedded Development Environment and how it will shape the future of Linux routers.

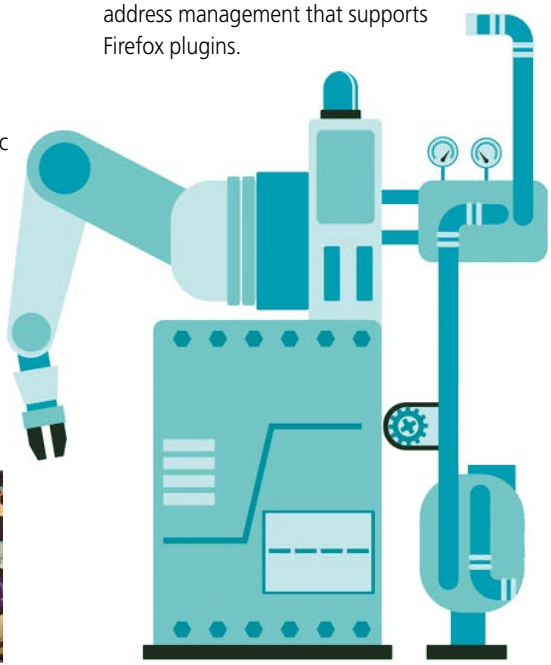
SERVICE

NEWS

REVIEWS

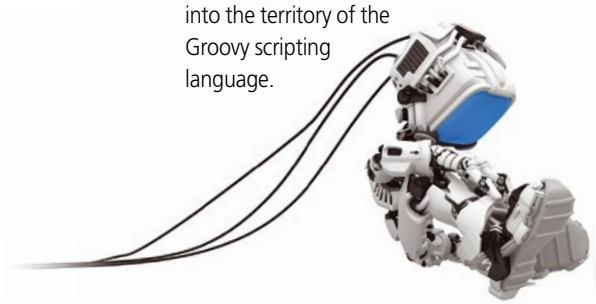


COVER STORIES

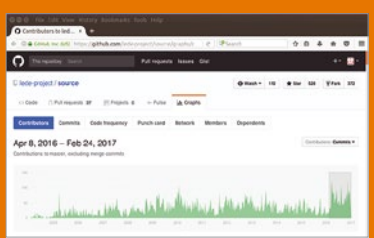


SEE PAGE 6 FOR DETAILS

IN-DEPTH



LINUXVOICE



On the DVD

TWO TERRIFIC DISTROS

DOUBLE-SIDED DVD!

Linux Mint 18.1 MATE (32-bit Live)

Linux Mint is a popular, Ubuntu-based desktop Linux with the emphasis on usability and out-of-the-box convenience. Linux Mint 18.1 "Serena" MATE Edition [1] [2] is a long-term support release, which means it will be supported until 2021. In addition to a number of bug fixes, updated translations, and new background art, the new edition continues the migration to GTK3 components. DuckDuckGo replaces the Google CSE search engine, and Rhythmbox replaces Banshee as the default music player. Other enhancements include a new search bar for the Xed text edition and several new features and updates for the Xplayer media player. The improved Update Manager now shows the source of updates and sorts them by origin.

AryaLinux Builder 2017 (64-bit Live)

AryaLinux, which bills itself as the "Linux for Builders" is a specialty Linux for users who wish to experiment with rolling their own Linux system. The newest 2017 Builder version is now 64-bit only [3] and includes build scripts to build the entire distribution from source code [4]. A builder script further simplifies the process: Only one script needs to be run, and advanced build options (e.g., configuring the kernel) can be specified while building the entire KDE or GNOME desktop environment without intervention from source. AryaLinux comes with kernel version 4.9 and boots into MATE desktop 1.17.

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

ADDITIONAL RESOURCES

[1] New features in Linux Mint: *https://linuxmint.com/rel_serena_mate_whatsnew.php*

[2] Linux Mint downloads: *https://linuxmint.com/download.php*

[3] AryaLinux release notes: *http://aryalinux.org/release-notes/aryalinux-2017.php*

[4] Building your own Linux from source using AryaLinux: *http://aryalinux.org/blog/2017/01/04/building-aryalinux/*

NEWS

Updates on technologies, trends, and tools

THIS MONTH'S NEWS

Dell Launches Five New Linux Systems

Dell seems to be the only major hardware vendor that continues to offer high-end Linux machines.

Barton George, Senior Architect in the office of CTO at Dell, wrote in a blog post, "Today I am excited to announce the next generation of our Ubuntu-based Precision mobile workstation line. Not only have we rev'd [revised] the current line-up but we have also added the Precision 5720 All-in-One."

The two new devices are the Dell Precision 3520 and Dell Precision 5520 mobile workstations. In addition to these two devices, three more Linux-powered devices that will be launched soon. The Dell Precision 7520 and Dell Precision 7720 mobile workstations will be available in March, and the Dell Precision 5720 all-in-one will be available in April.

All of these systems come with Ubuntu 16.04 LTS pre-loaded and feature seventh-generation Intel Core and Intel Xeon processors. Dell already sells a high-end Linux laptop: the Dell XPS 13 Developer Edition.

Microsoft Takes Skype for Linux Out of Alpha

Microsoft breathed new life into the almost defunct Skype for Linux in 2016 when the company released the alpha version of the rewritten client. However, the Linux client was behind its Mac OS and Windows 10 counterparts. Microsoft has now released the Skype for Linux 5.0 beta version, which inches toward bridging the feature gap.

A news item on Skype.com stated, "This update includes features that make it easier for you to use Skype for Linux for your everyday communications needs. We have been listening to you and added in some of your top requests."

Some of the new features of the beta include the ability to make phone calls to cellular phones and landlines using Skype Credits. Now users can also make one-to-one video calls from Linux to Mac OS, Windows, Android, and iOS and can view a shared screen from Windows and Mac OS clients.

Munich Plans to Ditch Linux and Go Back to Windows

The city of Munich is working on a plan to ditch LiMux, a customized Linux distribution, and go back to Microsoft products, including Windows.

The city of Munich has been using LiMux and other vendor-neutral technologies for more than a decade now. The move posed a serious challenge to Microsoft's dominance in the market. The success of LiMux would have encouraged other cities and regional governments to move away from Microsoft's proprietary technologies in favor of open source technologies.

Microsoft increased its lobbying efforts in Munich, which didn't see success under the previous administration. Microsoft's efforts started to pay off with the election of Microsoft-friendly Dieter Reiter as the mayor of the city, when the plans to ditch Linux and go back to Microsoft products returned to the table. In 2016, Microsoft opened a new headquarters in Munich to assist the new government in going back to the company's products.

Last year, the city of Munich conducted a study through a Microsoft partner, Accenture, which concluded that many departments are facing problems with Linux and open source technologies. In the month of February 2017, the city council came up with a plan to move to Windows by 2020.

Linux Kernel 4.10 Comes with New Security Features

Linus Torvalds has released Linux kernel 4.10, code-named Anniversary Edition, which brings many new features, including support for new hardware. One of the most significant improvements is support for virtual GPU. "This release adds support for Intel GVT-g for KVM (a.k.a. KVMGT), a full GPU virtualization solution with mediated pass-through, starting from 4th generation Intel Core (Haswell) processors with Intel Graphics," according to Kernelnewbies, "The capability of running native graphics driver inside a VM, without hypervisor intervention in performance critical paths, achieves a good balance among performance, feature, and sharing capability."

The Linux kernel supports many filesystems, and with this release, it has improved support for some of these filesystems, including ext4, F2FS, XFS, OverlayFS, NFS, CIFS, UBIFS, BeFS, and LogFS.

The release has also improved support for ARM-powered devices, including Huawei Nexus 6P (Angler), LG Nexus 5X (Bullhead), Nexbox A1 and A95X Android TV boxes, the Pine64 development board based on Allwinner A64, the Globalscale Marvell ESPRESSOBin community board based on Armada 3700, and the Renesas "R-Car Starter Kit Pro" (M3ULCB) low-cost automotive board.

Vlad Kochelaevskiy, 123RF

Announcing the release, Torvalds wrote on the Linux Kernel Mailing List (LKML): "On the whole, 4.10 didn't end up as small as it initially looked. After the huge release that was 4.9, I expected things to be pretty quiet, but it ended up very much a fairly average release by modern kernel standards. So we have about 13,000 commits (not counting merges – that would be another 1200+ commits if you count those)."

It's a big release, so it's not possible to list all of the new features here. Please refer to the Kernelnewbies writeup for detailed information.

OpenStack Day Program Announced

Several well-known speakers have committed to speak at OpenStack Day. Experts such as Alexander Stellwag and Kurt Garloff from T-Systems will speak about performance problems and the deployment model at OpenStack; Diederich Wermser will explain network automation in the OpenStack environment using SDN; and Wolfgang Ries and Carsten Duch will address the operation of an OpenStack installation. The complete program can be found online.

MORE ONLINE

Linux Magazine
www.linux-magazine.com

Off the Beat • Bruce Byfield

LibreOffice Drops Type 1 Font Support
Should decisions about free software be determined by the needs of developers or users?

Reglue Needs Your Support
Forget, for a moment, how open source seems to be the solution to business problems. Instead, return to basics and think of how free software helps to provide computers for the impoverished, and to reduce technological waste by extending the useful life of older hardware.

NTPsec: The Wrong Fork for the Wrong Reasons
Forks – the splitting of one project from another – are a natural part of free software. They are implicit in the Free Software Foundation's Four Freedoms, and I would no more attempt to deny the right of a fork to exist than I would attempt to insist that everyone use one Linux distribution or desktop environment.

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

Hierarchical Data Storage for HPC
Jeff Layton
I/O can be a very important part of any application. All applications need to read data and write data at some point with the possibility of huge amounts of both.

Modern Fortran for Today and Tomorrow
Jeff Layton
Fortran 90 took Fortran 77 from the dark ages by giving it new features that developers had wanted for many years and by deprecating old features – but this was only the start.

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

Halting the Ransomware Blackmail Wave
Thomas Gronenwald
In the tsunami of ransomware infections this year, the Locky encryption trojan is a high-water mark.

Setting Up MariaDB Replication with the Help of XtraBackup • Susanne Holzgraefe
If your database is so important that the content must not be lost between periodic data backups, replication is a possible solution.

The day will end with a panel discussion moderated by Linux-Magazin author Udo Seidel. Following the question "Quo vadis OpenStack?", well-known cloud experts will discuss the future of OpenStack.

The event is run by Computec Media GmbH, the publisher of *Linux-Magazin*, the German sister magazine of *Linux Pro Magazine*, in cooperation with event organizer Pluspol and Deutsche Messe AG. The main sponsor of the OpenStack Day is Telekom Deutschland GmbH.

Google Discloses Serious Security Flaws in Microsoft Products

Google disclosed an unpatched vulnerability in Internet Explorer and Microsoft Edge web browsers. After giving Microsoft 90 days to fix the bug, Google researchers have published the details of the vulnerability along with proof-of-concept code. The security hole affects all supported Windows versions, including Windows 7, 8.1, and 10.

According to The Hacker News, "The vulnerability (CVE-2017-0037), discovered and disclosed by Google Project Zero team's researcher Ivan Fratric, is a so-called 'type confusion flaw' in a module in Microsoft Edge and Internet Explorer that potentially leads to arbitrary code execution."

Google discovered the vulnerability in November 2016 and reported it to Microsoft on November 25, but for unknown reasons, Microsoft did not fix the problem. Google gave Microsoft three months to patch the security holes, and after that, Google publicly released the information about the vulnerability.

This is not the only security hole plaguing Microsoft's products. Earlier, Google disclosed flaws in Microsoft's GDI library that affects every Windows version all the way back to Windows Vista. Another unpatched flaw affects the SMB protocol, allowing attackers to crash the system.

Windows Helping the Spread of IoT Malware

Dr.Web, a cyber security firm, has found a Windows trojan that helps spread the infamous Mirai botnet across IoT (Internet of Things) devices. The newly found trojan targets Windows systems, and once installed, the trojan scans the network for connected IoT devices. If it finds a vulnerable device, it compromises the device and uses it in later attacks. Last year in October, Mirai brought down a huge chunk of the Internet by launching a DDoS (Distributed Denial of Service) attack on the Dyn managed DNS service.

The Windows trojan doesn't stop at compromising the IoT devices; it continues to spread itself to other Windows devices to further find and exploit more IoT devices.

Researchers noted that the malware could also identify and compromise database services running on various ports, including MySQL and Microsoft SQL, to create a new admin *phpminds* with the password *phpgodwith*, allowing attackers to steal the database. At this time, it's not known who created this trojan, but the attack design demonstrates that IoT devices that are not directly accessible from the Internet can also get hacked to join the Mirai botnet army.

IoT devices are already vulnerable to infection, so why are malware writers targeting Windows? Primarily because Windows still dominates the market and it gives the malware writers another platform to spread the botnet.

This is not the only security hole plaguing Microsoft's products. Earlier, Google disclosed flaws in Microsoft's GDI library that affects every Windows version all the way back to Windows Vista. Another unpatched flaw affects the SMB protocol, allowing attackers to crash the system.

OSDC

OPEN SOURCE DATA CENTER CONFERENCE

MAY 16 – 18, 2017
BERLIN

HIGHLIGHT SPEAKERS

Mathias Meyer
CEO | Travis CI

Seth Vargo
Director of Technical Advocacy | HashiCorp

Mandi Walls
Technical Practice Manager EMEA | Chef

MAIN TOPICS

CONTAINERS & MICROSERVICES

CONFIGURATION MANAGEMENT

TESTING, METRICS & ANALYSIS

TOOLS & INFRASTRUCTURE

www.osdc.de

NETWAYS®

Zack's Kernel News

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

Revamping the Firmware API

Luis R. Rodriguez didn't like that the kernel's firmware API had been gradually extended into a nightmare mash of hopeless deformity. Any time someone updated a firmware routine, or added a new one, all existing firmware users had to be sifted and updated to account for the change. The API had also been extended to the point that user code had begun to use it for things other than firmware. As Luis put it, "an example here is the p54 driver enables users to provide a custom EEPROM through this interface. Another example is optional CPU microcode updates. This list is actually quite endless." He added that there was a queue of other subsystems wanting to use the firmware API for their own non-firmware purposes.

Luis proposed, first of all, continuing to support the existing firmware API. Beyond that, it would provide a superset of features, some of which could be used for firmware, but that would have more general "data helper" value as well. The new API would handle calls synchronously or asynchronously as needed, and in general the API would be extensible in natural ways that would not require going back to change all existing usage every time a new extension arrived.

Greg Kroah-Hartman had no major objections and mostly offered documentation suggestions.

One thing Luis's code did not do was address the firmware fallback options. If a firmware update doesn't work, it's nice to be able to use a previous version while you replace the new version. But Luis found the fallback code to be "hairy" and wanted to tackle it properly. So his initial plan was to leave all fallback fixes out and just deal with them as soon as possible. As he put it, "I really am trying hard to make that fallback mechanism *work* fine; right now its just hairballs."

On another note, Luis added, "There are also some more longer term things to consider which I'd like to address as well, for instance, the firmware code is the only code using the general UMH [User Mode Helper] lock, although it was originally added to help warn for firmware API uses on suspend/resume, the firmware cache mechanism now helps resolve the issues – the only case which cannot take advantage of this is the custom fallback mechanism. One also needs to consider if the UMH lock or a replacement should be considered for other kernel UMH."

So Luis would not be fixing absolutely everything all at once. But he also pointed out that there were currently

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.**

race conditions with respect to the kernel's init code, that couldn't really be addressed properly with the old firmware code still in place. Once the new code had been adopted, he said, the races would be much more fixable.

Bjorn Andersson asked for some reassurance that when the new fallback design did get coded into the kernel, it would be compatible with existing user space, "or will we forever have two duplicate systems for loading 'firmware' in the kernel?"

Luis replied that the old API would be sticking around indefinitely. But that it would become static, and all new features would be added to the new API. Beyond that, he said, he wasn't certain exactly how compatible the old and new APIs would be, and so he couldn't say for sure that the old API would definitely go away.

I love reading about these crazy rescue missions. It's always something everyone relies on, but that hasn't been maintained in years and has just been accruing patches like crustaceans until the whole thing is ready to tip over and fall back into the sea. Except someone decides to write a whole new thing that's not quite as susceptible to crustacean infestation.

Crypto Optimization

Binoy Jayan wanted to optimize some of the kernel's cryptographic routines. Specifically, he pointed out that the initialization vectors (IVs) were generated in the `dm-crypt.c` file. If they could be implemented as template ciphers in the kernel's cryptographic layer, the IVs could run directly in hardware, rather than being implemented in software. This would produce a significant speedup, he said.

With the IV generator code migrated, the dm-layer code could be optimized to send a whole "bio" at a time, and Binoy explained that "each bio contains the in memory representation of physically contiguous disk blocks. The dm layer sets up a chained scatterlist of these blocks split into physically contiguous segments in memory so that DMA can be performed."

Milan Broz cautioned against Binoy's approach. Some of the IV generator code, he said, were "hacks," and some existed only for compatibility reasons and should not be regarded as safe. He also said that if the IV generator code was moved into the kernel's crypto layer, it would make it difficult to change the cryptographic key structure later on, if they should ever want to.

Herbert Xu felt that the code could remain in the `dm-crypt.c` file under the control of the dm-crypt team but still be registered in the kernel's crypto layer to take advantage of the greater hardware efficiency.

Herbert and Binoy went back and forth for a bit on implementation issues and came to some kind of understanding about how to divvy up different pieces of the code.

Meanwhile Gilad Ben-Yossef was not even able to test Binoy's patches because they wouldn't apply cleanly against his tree. Binoy explained that "there were some key structure changes in dm-crypt after I sent out v2. I have resolved them while working on v3. Please wait for the next version of the patchset. I'll send it probably by next week. I wanted to incorporate a few changes suggested by Herbert before sending them."

At the same time, Ondrej Mosnacek came in with his own objections:

"I like what you are trying to achieve; however, I don't think the solution you are heading towards (passing sector number to a special crypto template) would be the best approach here. Milan is currently trying to add authenticated encryption support to dm-crypt and as part of this change, a new random IV mode would be introduced. This mode generates a random IV for each sector write, includes it in the authenticated data and stores it in the sector's metadata (in a separate part of the disk). In this case, dm-crypt will need to have control over the IV generation (or at least be able to somehow retrieve it after the crypto operation).

That said, I believe a different approach would be preferable here. I would suggest, instead of moving the IV generation to the crypto layer, to add a new type of request to [the] skcipher API (let's call it `skcipher_bulk_request`), which could be used to submit several messages at once (together in a single sg list), each with their own IV, to a skcipher. This would allow drivers to optimize handling of such requests (e.g., the SIMD ciphers could call `kernel_fpu_begin/end` just once for the whole request). It could be done in such a way that implementing this type of requests would be optional and a fallback implementation, which would just split the request into regular `skcipher_requests`, would be automatically set for the ciphers that do not set it themselves. That way, this would require no changes to crypto drivers in the beginning and optimizations could be added incrementally.

The advantage of this approach to handling such 'bulk' requests is that crypto drivers could just optimize regular algorithms (xts(aes), cbc(aes), etc.) and wouldn't need to mess with dm-crypt-specific IV generation. This also means that other users that could potentially benefit from bulking requests (perhaps network stack?) could use the same functionality."

Mosnacek added that he was currently working on a set of patches to implement a proof-of-concept for this idea and planned to post something soon. But at this point, the conversation came to an end.

This discussion yielded no clarity about how things would eventually work out. Which is great – it's just folks talking about how to make something cool. What I like about these kinds of discussions is how people from disparate areas of the kernel bring what they know, often in ways that surprise each other, and end up coalescing around a solution none of them could have envisaged at the start.

Fixing Memory Access for PCI Devices

Nikita Yushchenko pointed out that a given PCI device could conceivably support up to 64-bit direct memory access (DMA) addressing. But PCI host bridge had limitations on its addressing abilities that prevented it from accessing that much RAM. Nikita felt that instead of allowing PCI devices to claim the ability to access 64-bits when in practice they wouldn't be able to, it would be better to prevent them from claiming that ability in the first place.

He posted a patch to implement that. Will Deacon objected that the patch treated the problem as if it were architecture-specific, which it wasn't. He suggested that "another hack you could try would be to register a PCI bus notifier in the host bridge looking for `BUS_NOTIFY_BIND_DRIVER`, [and] then you could proxy the DMA ops for each child device before the driver has probed, but adding a

`dma_set_mask` callback to limit the mask to what you need."

Arnd Bergmann posted an experimental patch of his own, saying, "This is what I prototyped a long time ago when this first came up. I still think this needs to be solved properly for all of ARM64, not with a PCI-specific hack, and in particular not using notifiers."

Nikita objected that Arnd's patch was far from complete and added, "In current device trees no dma-ranges is defined for nodes that are parents to PCI host bridges. This will make `of_dma_configure()` to fall back to 32-bit size for all devices on all current platforms. Thus applying this patch will immediately break 64-bit DMA masks on all hardware that supports it."

Arnd disagreed with Nikita's diagnosis, saying that the patch wouldn't break 64-bit DMA masks. But he also felt that it wasn't clear where the true underlying kernel issue was. He asked, "Is it actually the PCI host bridge that limits the ranges here, or the bus that it is connected to?"

Nikita acknowledged that he wasn't sure where the true source of the limitation was. He and Arnd dove into a technical exploration of the problem. At one point, Arnd remarked, "we have to guarantee that the fallback to 32-bit DMA always succeeds. There are also a lot of drivers that try to set a 64-bit mask but don't implement bounce buffers for streaming mappings if that fails, and swiotlb is what we use to make those drivers work."

(He also added, "And yes, the API is a horrible mess." And Nikita replied, "The entire infrastructure to allocate and use DMA memory is messy.")

At one point, Arnd gave an historical analysis of the situation. "What I think happened here in chronological order is:

- In the old days, 64-bit architectures tended to use an IOMMU all the time to work around 32-bit limitations on DMA masters.
- Some architectures had no IOMMU that fully solved this and the dma-mapping API required drivers to set the right mask and check the return code. If this failed, the driver needed to use its own bounce buffers as network and scsi do. See also the grossly misnamed `PCI_DMA_BUS_IS_PHYS` macro.
- As we never had support for bounce buffers in all drivers, and early 64-bit Intel machines had no IOMMU, the swiotlb code was introduced as a workaround, so we can use the IOMMU case without driver-specific bounce buffers everywhere.
- As most of the important 64-bit architectures (x86, arm64, powerpc) now always have either IOMMU or swiotlb enabled, drivers like NVMe started relying on it, and no longer handle a `dma_set_mask` failure properly."

At some point, Christoph Hellwig joined in the fun, and he and Arnd continued the technical analysis. Eventually, the thread petered out inconclusively.

Ultimately, no clear diagnosis of the problem came out of the discussion, but it's clear that the problem exists and the current code is broken. So that's something. It's also clear that several folks are delving into the code to try and untangle what went wrong, so there will probably be some kind of fix in the near future.

Extending Boot Support

Chao Peng said:

"Multiboot specification is an open standard that provides kernels with a uniform way to be booted by multiboot-compliant bootloaders (like grub).

This patch is trying to make [the] Linux ELF kernel image to be a multiboot-compliant OS so that it can be loaded by a multiboot-compliant bootloader. The benefit is eliminating the maintenance for realmode and decompression code and especially when the kernel is loaded in a virtual machine, the reducing for these code can greatly cut down the boot time."

Unfortunately, he said, the specification wasn't completely clear about how to handle 64-bit kernels. Also, in spite of the grub program supporting the ELF64 format, it was currently unable to read the memory address required to load the program.

But H. Peter Anvin put his foot down in a one-line email, saying, "As has been shown many times before, this is a really bad idea. Unless there is a real-life use case where this matters enormously, this is nacked with extreme prejudice."

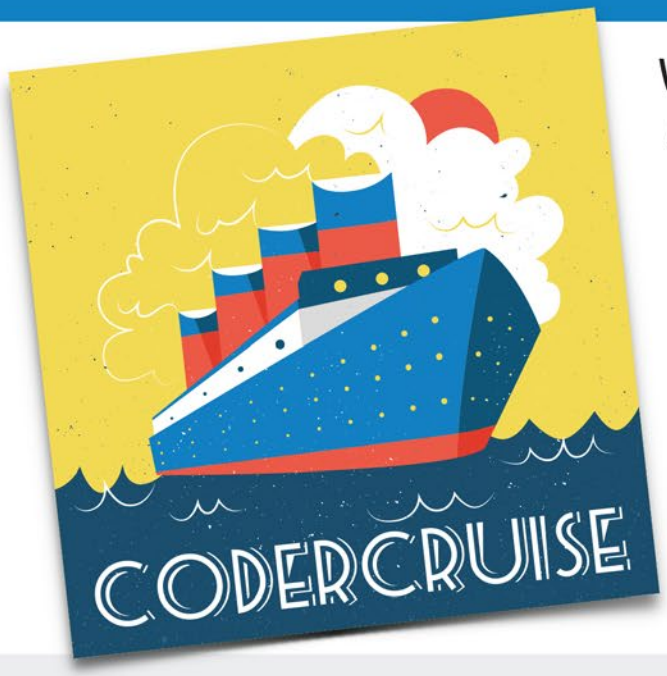
As an alternative, Luis R. Rodriguez remarked, "something to consider, provided the issues with multiboot get resolved: If you want to boot Xen, you actually use the multiboot protocol, the last PVH boot patches had borrowed ideas from Multiboot to add an entry to Linux, only it was Xen'ified. What would be Multiboot 2 seemed flexible enough to allow all sorts of custom semantics and information stacked into a boot image. The last thought I had over this topic (before giving up) was – if we're going to add yet-another-entry (TM) why not add extend Multiboot 2 protocol with the semantics we need to boot any virtual environment and then add Multiboot 2 support entry on Linux? We could redirect any custom boot mechanism then to just use that given its flexibility."

H. Peter clarified his objection, saying, "Multiboot has a fundamentally broken assumption, which is to do certain work for the kernel in the bootloader. This is fundamentally a bad idea, because you always want to do things in the latest step possible during the boot process, being the most upgradeable, and have the interface as narrow as possible. Therefore, using Multiboot is actively a negative step. It is declared an 'Open Standard' but anything can be such declared; it really is a claim that 'everything should work like Grub'."

Daniel Kiper, creator of Multiboot2, took umbrage to H. Peter's criticism, and replied, "I can agree that they are not perfect (especially Multiboot proto is very inflexible). However, both protos try to standardize boot process. I think it is nice because right now almost every (new) kernel has [its] own boot protocol (some even support more then one, sic!). And it is [an] enormous task to support all of them in one boot loader. So, I think that [the] Multiboot protocols family (IMO, Multiboot2 is preferred today) are [a] good idea. Are they not perfect? Yes, but I do not think that proliferation of tons of incompatible boot protocols, each specific for one kernel, is better. So, if you think that we can fix something in Multiboot2 please tell us. If you think that it is unfixable, please tell us too. We can think about Multiboot3 too (ehhh… maybe this is not the best idea). Anyway, it would be nice if one day we have one common boot protocol for (almost) everybody."

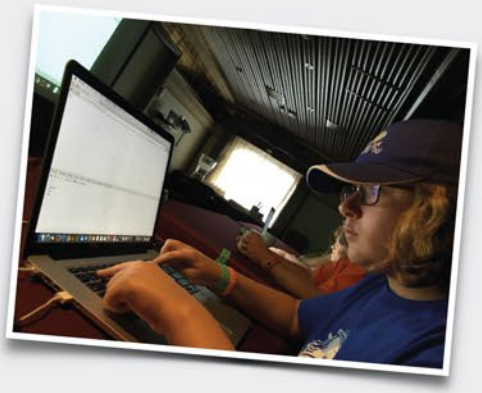
The discussion looked like it might get violent, but instead it trailed off inconclusively. It seems clear that support for ELF64 is one of those features where you think you're just implementing a cool feature, but actually you're stepping into a lake of boiling lava. It seems that the underlying issues of boot protocols need to find some resolution, before developers will be able to comfortably add certain new boot-time features. ■■■

A Conference on a Cruise Ship?



We hope your initial reaction was ***HECK YEA***! Announcing CoderCruise — a conference for people who code & their families — focused on web technology. It doesn't matter what your preferred programming language is; we have sessions that will appeal to you in this unique experience.

This event isn't just family friendly, your **entire family** is included in the conference for **free**. We have sessions planned to teach kids (and other family members) how to code!



With **33 sessions** in two tracks, taught by 26 experienced professionals, you can be sure there will be plenty to interest you. Conference tickets are just ***$295*** and the cruise starts at $882. Contact us for bulk pricing to bring your whole team for an amazing team building experience that you won't forget!

Use code: CC-DWD-AMN to save 15%

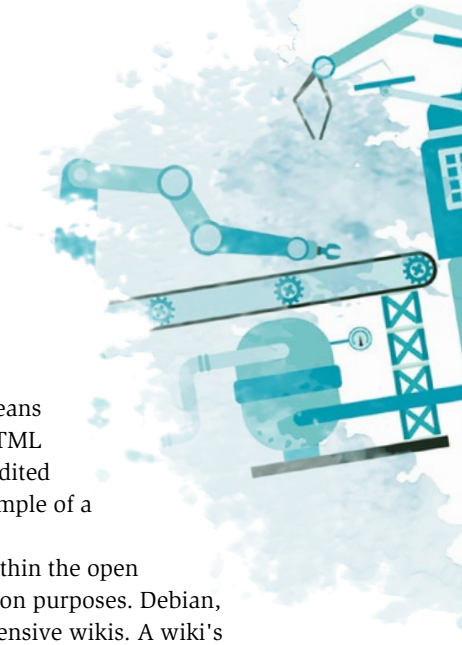
July 16-23 — New Orleans

www.codercruise.com

Zim, the cross-platform desktop wiki

Daily Diary

Organize your ideas, notes, and shopping lists with Zim, a handy tool that enshrines the principle of the wiki on your Linux, Mac OS, or Windows desktop. *By Ferdinand Thommes*



The word wiki [1] comes from Hawaiian and means "fast." The term refers to a system in which HTML documents on the Internet are both read and edited using a web browser. The most prominent example of a wiki is the Wikipedia online encyclopedia.

Other wikis inhabit both large and small projects within the open source universe. Many wikis are used for documentation purposes. Debian, Ubuntu, and Arch Linux are known for their comprehensive wikis. A wiki's main task is to organize content in a structured way. Many of the popular wiki tools are designed for collaboration over the network, but the wiki concept also provides benefits for a single user working at a single desktop. Zim is a handy desktop wiki that is useful for brainstorming and knowledge gathering, as well as building task lists and organizing documents and books.

Versatile

Zim [2] transfers the principle of a wiki to the desktop. To operate Zim, you do not need a web server or a database (Figure 1). The Zim desktop wiki is available for Linux, Mac OS X, and Windows; it uses the same data format across all platforms. If you are looking for something to compare to Zim, a tool from the Windows world that also runs on Linux, such as WikidPad [3], is conceptually similar, but Zim goes well beyond WikidPad's capabilities.

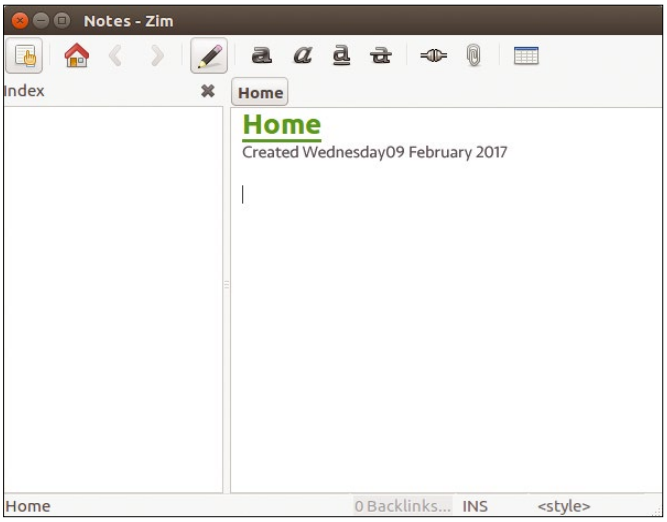


Figure 1: A first glance reveals little about the many capabilities of the Zim desktop wiki.

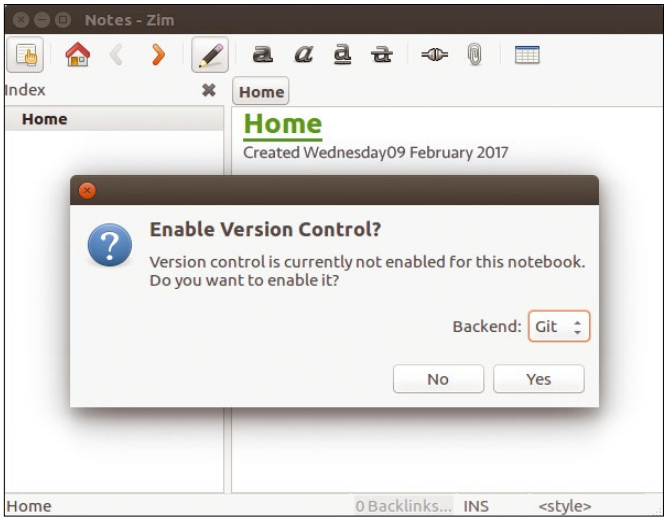
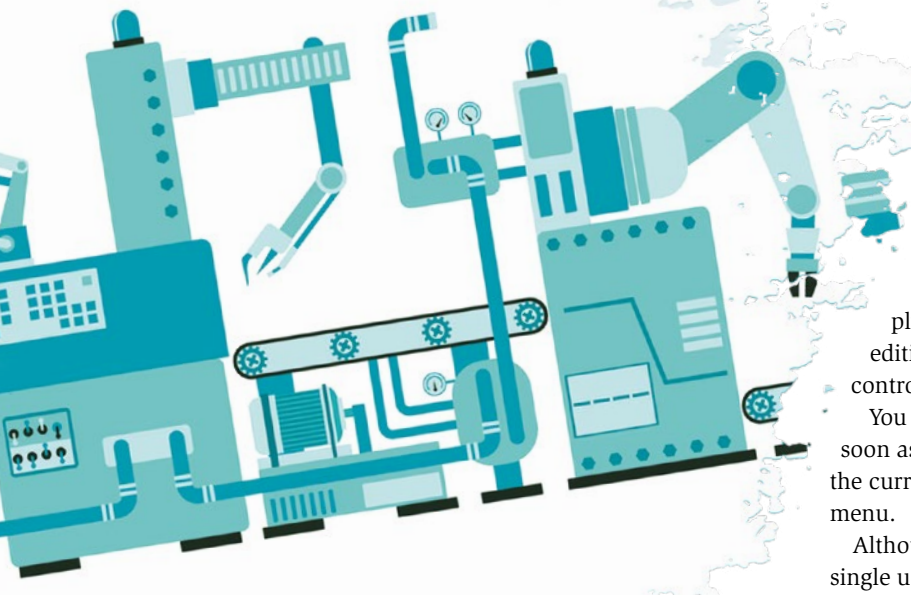


Figure 2: The built-in Zim version control system is enabled in the File menu.



Zim is based on the GTK framework and resides in the repositories of most Linux distributions.

Markup and Versioning

Every page created in Zim is stored on disk as a text file with a wiki markup. The program organizes these files into notebooks, which can best be compared with the database of traditional wikis.

To create a new page, first create a link to the initially non-existent page. You can then add references to other sites to the text of the page. Zim also supports plugins, including one for versioning, which stores the editing history for a document in the form of the version control systems (VCSs) Git, Bazaar, or Mercurial (Figure 2).

You need to install the VCS locally on the computer. As soon as you enable the plugin via the settings, you can store the current version of a document using the *File* | *Save version* menu.

Although Zim is intended for local use and focused on the single user, the versioning feature does provide an option for collaboration. The VCS structure is also shown in file managers that display version controlled documents separately, such as Dolphin or Files, the file manager formerly known as Nautilus (Figure 3).

Extensible Through Plugins

You can extend the Zim desktop wiki using around 30 preinstalled plugins. On top of this list, you will find numerous plugins by external developers on GitHub [4]. These plugins include support

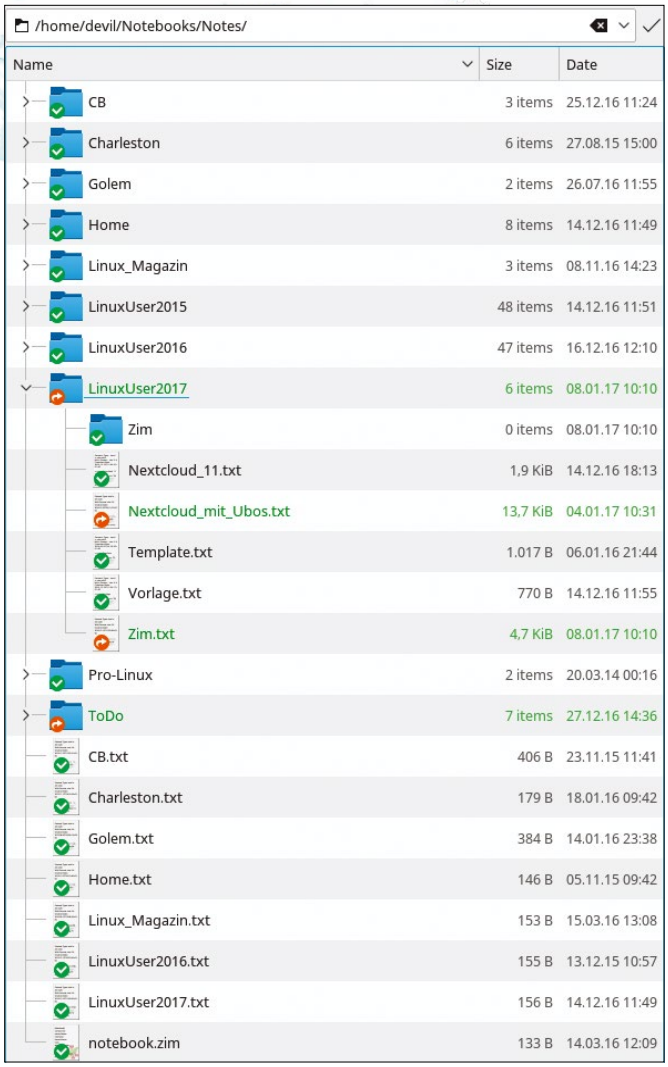


Figure 3: The view in Dolphin, using Git to show version control in the notebooks.

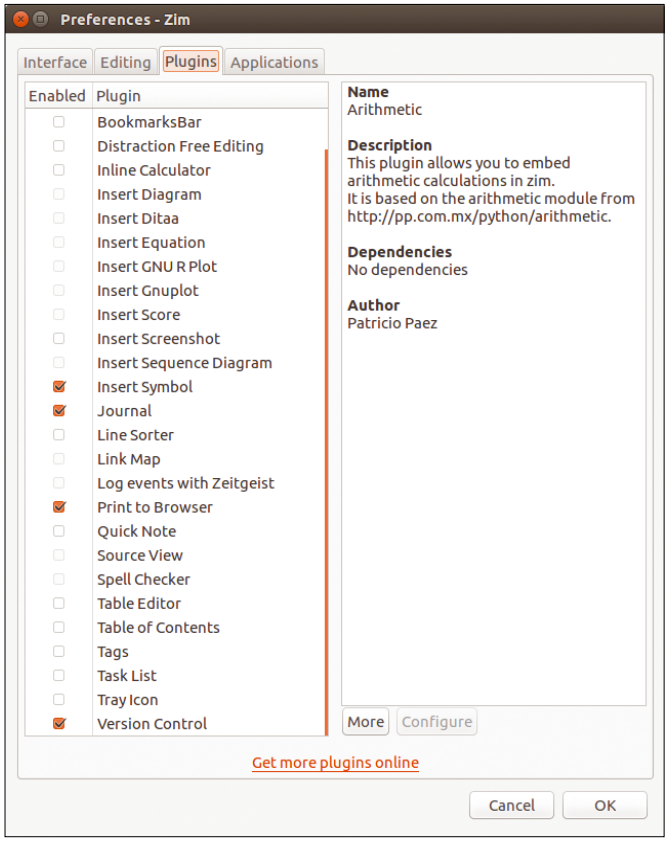


Figure 4: You can extend Zim using numerous internal and external plugins.

for formulas, charts, and music notation, plus a source code view and a spell checker. Zim can "print" to an HTML file and has a table editor, keyword management tool, and calculator (Figure 4).

You can format text in various ways and supplement it with pictures, lists, enumerations, and annexes. In the *Tools | Custom tools* menu, Zim gives users the opportunity to create their own extensions. In addition, you can launch your own web server in the Tools menu and thus view your Zim wiki in the browser.

First Launch

When you launch Zim, it asks you where to store the data. Zim then creates a first notebook at the defined location, names it *Home*, and adds the `.txt` suffix. *Home* is then opened as a notebook whenever you launch Zim.

In theory, you could now just start typing in Zim's WYSIWYG editor; however, it makes sense to define the structure of your wiki first. For each project (say, a book project, a collection of links, or a to-do list), you will first want to create a separate notebook to provide an easy means for distinguishing the content.

Simple Formatting

A look at the Format menu clarifies the formatting that you can apply to the currently selected text via the toolbar or the usual keyboard shortcuts. If you do not select a text passage explicitly, Zim formats the word under the cursor. In addition to the usual markup features, such as bold, italics, underline, and strikethrough, you will find five levels of headings, and you can also highlight words or text passages in yellow.

Highlight source code by means of a fixed-width font to set it apart from the remaining text. Charts, lists, and enumerations complete the picture. If you use the right plugin, you can also embed calculations in Zim and represent formulas. The same applies to Gnuplot graphics and images. You can also use an arbitrary image editor while working within Zim; launch the image editor via the Context menu of an image and save the results in Zim. The original image is not changed.

Write Protection and Journal

In the toolbar is a symbol with three dots and a pen. If you hover the mouse pointer over this symbol, the *Edit notebook* label appears. Clicking this label enables write protection for the entire notebook. In the calendar, clicking on the icon labeled *Today* in the active notebook generates the structure of a journal with sub-items for the year, month, and day that you can use as a diary.

While you are working, Zim continuously stores the current state. Finished documents can then be exported as HTML, LaTeX, or Markdown (Figure 5). The tab bar below the toolbar, which grows to accommodate the increasing number of projects, shows the state of editing and cannot be deleted during the session.

Lists, Links, and Annexes

If you start a line with an asterisk (`*`) or square brackets (`[]`), Zim automatically switches to a mode in which it is very easy to create bullet lists or checklists (Figure 6). To add a little more structure, indent the lines using the Tab key. Zim continues these lists until you insert a blank line.

If you enable the *tasklist* plugin, you can expand on this function. Supplement the bullet by typing, say

```
[] Task 1 02-23-2017 !
```

to include a due date and a priority (each additional exclamation mark decreases the priority), and then display all the tasks of the project by pressing the *Task List* button in the toolbar.

Internal links to other pages of the notebook you are currently editing, or to another notebook, mean that Zim also has a powerful organizational function. You can either press Ctrl + L to create a reference to a non-existent page (Zim then creates the page automatically), or link to an existing page. The link appears in blue in the text.

Alternatively, you can create links without dialogs or shortcuts using CamelCase, or WikiWord notation [5]. For example, entering "LinuxMagazine" automatically generates a link of the same name and the appropriate page due to the uppercase "M." Since this feature can lead to unwanted links, you have the option of switching off CamelCase linking through the configuration settings.

Zim automatically converts paths such as `/etc/default` into links; click-

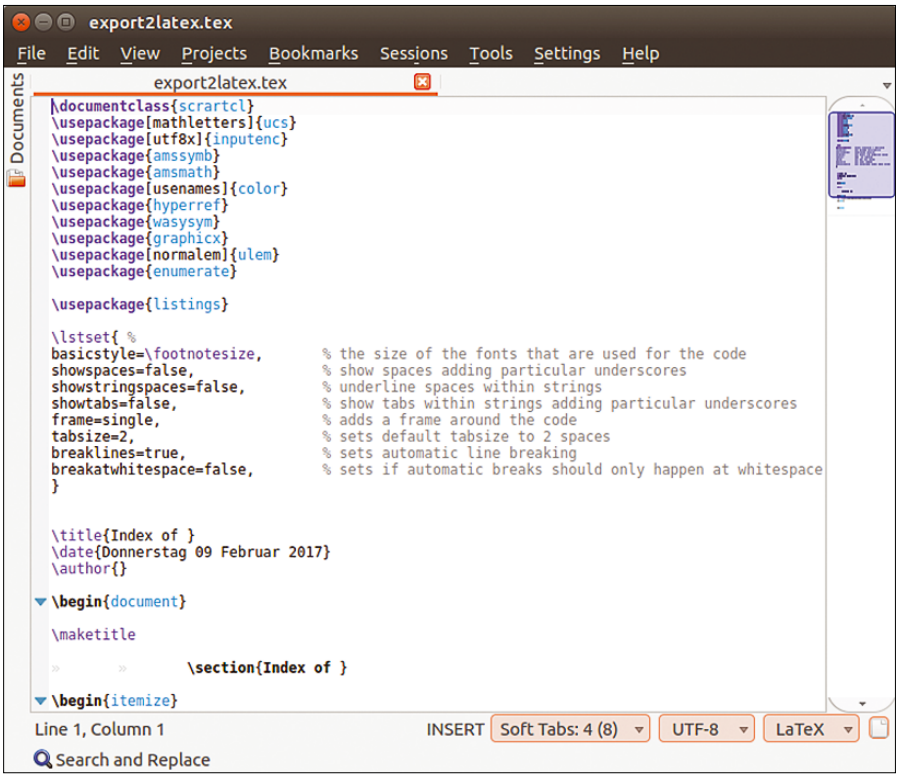


Figure 5: Kate displaying an article exported from Zim in the universal LaTeX format.

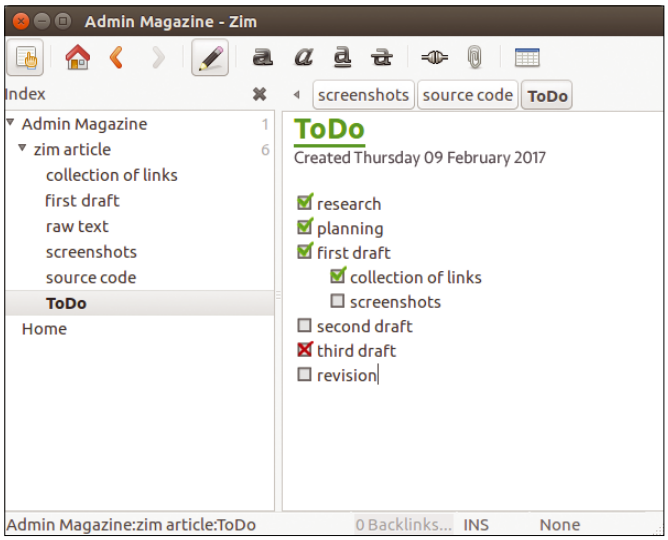


Figure 6: The organization of this article as a task list in Zim.

ing on one of these links takes you to the appropriate directory or opens the associated file. Local files can be saved as files attachments using the *Tools | Attach* menu item. Zim assigns the attachments to the article and moves them to the new instance if you copy or move the original. If you copy the wiki to a second computer, the attachments remain in place.

Conclusions

Zim is a very flexible tool that lets you store information in a structured way and continue processing it with other applications. The program is written in Python and released under the GPL. The documentation on the project page [6] illuminates all aspects of the application in detail and also provides a page specifically for newcomers [7].

The data is stored in text format with wiki markup, thus opening up numerous possibilities. For example, you can use a make-file to generate a website from the wiki entries. Thanks to the text format with wiki markup, you can use the stored information on all three supported operating system platforms. The flexible methods for linking let you build rich documents that you either use as a preliminary stage for processing in other applications or finalize in Zim.

Zim is easily accessible, revealing the power of its options only when needed. Once you have discovered the versatility of Zim, you will not want to be without this desktop wiki, which has been in development since 2005. Even Linux beginners who are familiar with Microsoft OneNote are likely to quickly find their way around Zim.

The Android version of Zim [8] is currently a work in progress. The data exchange will be via platforms such as Nextcloud or Dropbox. Additionally, you will find a portable version of Zim, suitable for use on a USB stick, at PortableApps [9]. Zim impresses as a complete package, and if you like the organizational structure of wikis, you are bound to feel right at home. ■■■

INFO

[1] Wiki: *https://en.wikipedia.org/wiki/Wiki*

[2] Zim: *http://www.zim-wiki.org*

[3] WikidPad: *https://en.wikipedia.org/wiki/WikidPad*

[4] Plugins: *https://github.com/jaap-karssenberg/zim-wiki/wiki/Plugins*

[5] CamelCase: *https://en.wikipedia.org/wiki/Camel_case*

[6] Documentation: *https://github.com/jaap-karssenberg/zim-wiki/wiki*

[7] Getting Started: *http://zim-wiki.org/manual/Usage/Getting_Started.html*

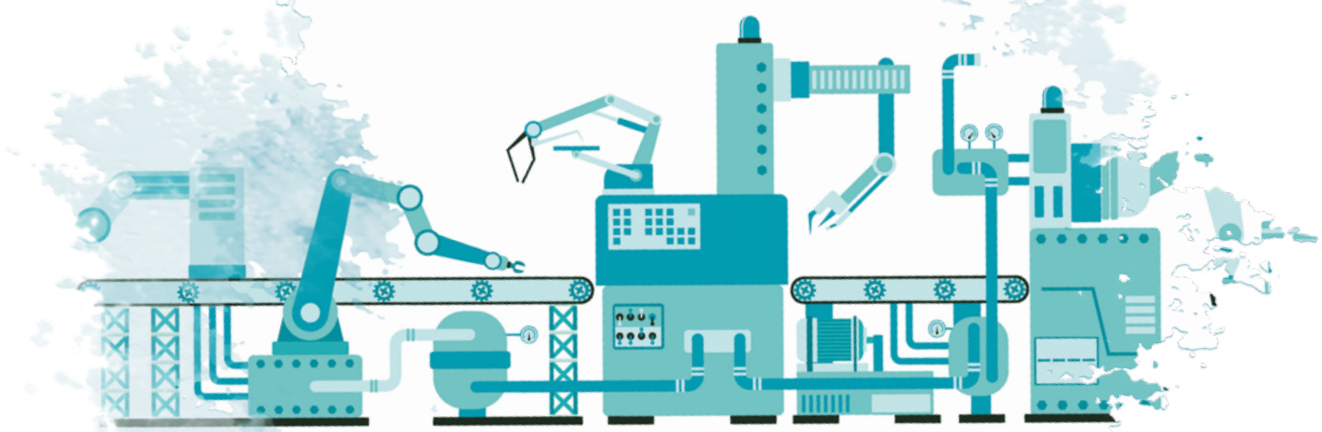
[8] Android app: *https://github.com/jaap-karssenberg/zim-android-mockapp/wiki*

[9] Portable Zim: *http://portableapps.com/node/16979*

Introducing the emerging office suite, OnlyOffice Desktop Editor

Office Newbie

The OnlyOffice Desktop Editor is a free office suite built for collaboration. *By Erik Bärwaldt and Christoph Langner*



The Linux environment is home to several practical and high-functioning office suites. Many mainstream distros come with LibreOffice pre-installed, but other alternatives, such as SoftMaker Office [1], WPS Office [2], Apache OpenOffice [3], and the Calligra suite [4] also compete for Linux mindshare.

The Linux version of OnlyOffice Desktop Editor (formerly Teamlab Office) [5] is a recent arrival to the Linux scene. OnlyOffice is developed by a proprietary company, but a free and open source version of the suite is available and licensed under the GPL.

Like other leading office suites, OnlyOffice comes with a word processor, spreadsheet, and presentation tool. The focus is on collaborative work in a professional setting, but OnlyOffice also works well on a single workstation. Is OnlyOffice better than what you're using now? It depends on your taste and preferences, but the OnlyOffice suite does offer some interesting innovations, such as a free community server for interactive access in a local cloud environment.

Setting Up OnlyOffice

OnlyOffice is developed and maintained by the vendor Ascensio System. The software is available in server and desktop versions. The server installation (as a free Community Edition and commercial Enterprise Edition) is aimed at businesses or workgroups that want to use the office package as a web application.

To install on a single computer, select the desktop version, which is available in the form of DEB and RPM packages for 64-bit architectures only. A portable version in

LISTING 1: Installing the Debian Package

```
01 $ sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv-keys CB2DE8E5
02 $ echo "deb http://download.onlyoffice.com/repo/debian squeeze main" | sudo tee -a /etc/apt/sources.list
03 $ sudo apt update
04 $ sudo apt install onlyoffice-desktopeditors
```

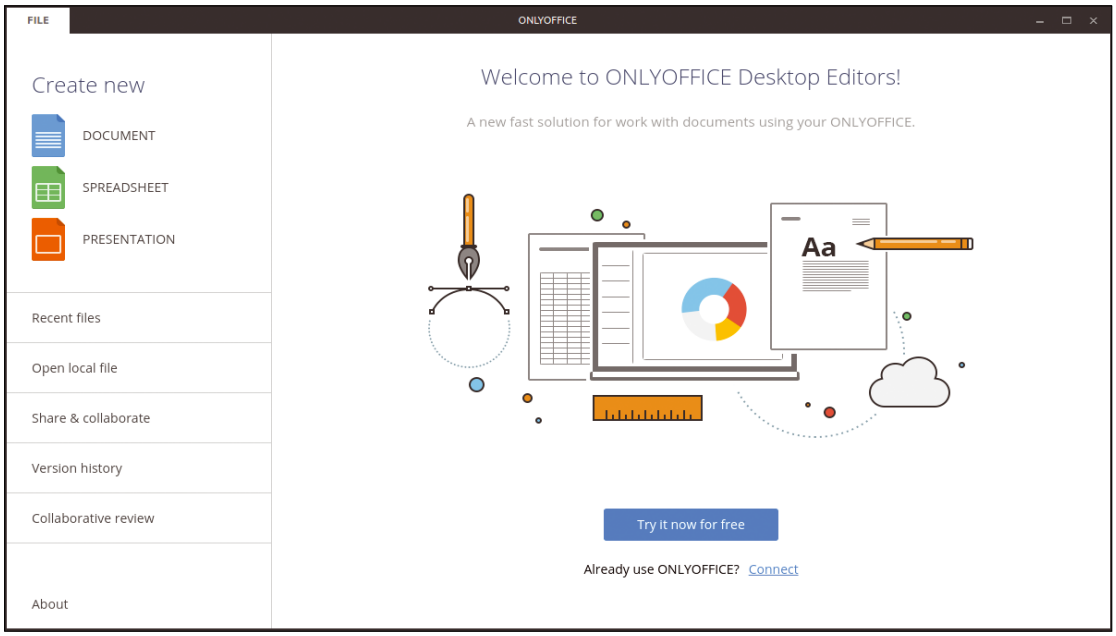


Figure 1: The splash screen in OnlyOffice at first reminds you of LibreOffice.

the form of a tarball is also available. The size of the installation packages varies between 170 and 250MB, which is not uncommon for office suites. Alternatively, the vendor maintains a package repository for Debian or Ubuntu systems (Listing 1).

Use the package manager for your Linux distribution to install the DEB or RPM packages. To install the portable version, first unpack the tarball using your archive manager, or enter:

```
tar xzf onlyoffice* .tar.gz
```

in a terminal window. To start the suite, change to the newly created subdirectory `desktopeditors/` and call `./onlyoffice-desktopeditors.sh`.

After the first call, an overview window appears that is vaguely reminiscent of the LibreOffice splash page: Top left in this two-pane window; you can click on *Document*, *Spreadsheet*, or *Presentation* to create a new file (Figure 1). Below are the options *Recent Files* and *Open local file*, which let you access or create locally stored documents.

The *Share and collaborate* option takes the user to functions for collaborative work in the cloud. You can create a portal in the OnlyOffice cloud (it is free for 45 days), or you can connect to your own server (but more of that later). The server does not need to reside on the Internet; you can simply use OnlyOffice on your local network.

Interface

When you create a new document with OnlyOffice, the first thing you'll notice is the unusual interface: Unlike mainstream alternatives like LibreOffice and SoftMaker, OnlyOffice is oriented on the look of newer Microsoft Office versions. You will thus find the many icons arranged in two rows; a menu bar is missing. On the left and right of the document are even more controls. This design might make the mouse travel longer than necessary for many actions, but it does use the screen space provided by a full-HD display in a better way.

OnlyOffice lets you keep multiple documents open simultaneously in a window, organizing the files in horizontal tabs at the top of the screen. This design lets you easily toggle between active documents. Unlike in the traditional office suites, the settings menu is limited to just a few choices, with hardly any customizing options (Figure 2). You can access the settings by clicking on the gear icon in the top right corner of the window.

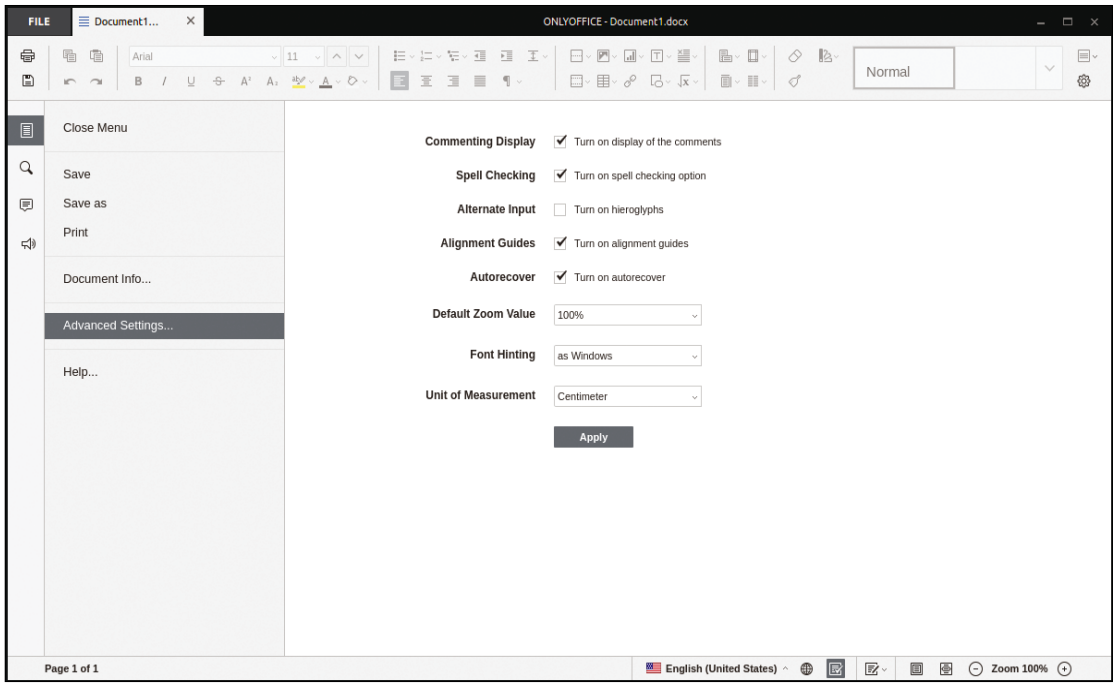


Figure 2: Even the advanced settings in OnlyOffice offer only a few configuration options.

Format Questions

OnlyOffice saves new documents in Microsoft's .docx format by default, but the suite also can handle other formats: Besides simple text without formatting, OnlyOffice supports the Open Document specification, as used by the free office suites. In addition, documents can be stored as RTF or PDF documents.

To change the format of an existing file, click on the *File* button next to the document box, and then select *Save as*. OnlyOffice also does its own thing when you open an existing document: There is no menu item for this purpose; instead you always need to open existing files from the *File* menu in the main window. Alternatively, you can drag and drop documents into the application window.

A Question of Settings

OnlyOffice only provides rudimentary options for modifying the functions and appearance of the software. The settings are also spread across different dialogs.

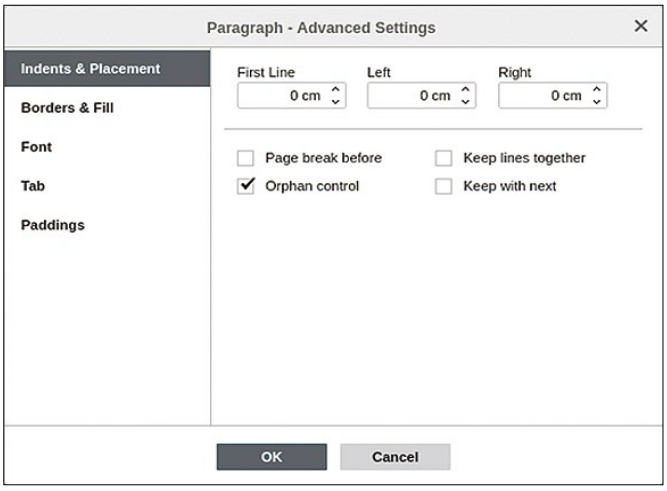


Figure 3: OnlyOffice is miserly with the options it offers, even for simple things such as text formatting.

The *Advanced settings* gear icon, which is found top right in the window in all three program modules, only gives access to basic customization options. But to customize the toolbars, you need to click on the tray icon above it for the *View settings*. This opens a small context menu where you can change the appearance of each module.

Options for text editing, on the other hand, are found in the *Paragraph settings* in the right corner of the application window. But you need to press the *Show advanced settings* button to open another window where you can actually configure the settings (Figure 3). Additional settings for the currently active file are found in the *File* dialog on the left border of the application window. Again, the options are limited to the absolute minimum.

A speech bubble icon that activates comments is available on the left edge for collaborative work. Clicking on the icon reduces the document area. In the new view, you can then press the *Add comment to document* button to annotate the document. The *Add response* button then lets you add a reply in this part of the window. The comments are displayed clearly, one under another, so the recipient has an overview of the entire history.

Cloudy

OnlyOffice provides seamless interfaces to proprietary cloud services, as well as to the local server, which you can operate autonomously. In the desktop variant, the *Connect to server* menu item in the main window is the entry point to the portal entry and acts as a platform for collaborative work – assuming you have a matching account.

The office package then loads a web-based editor in the existing interface that has the same functions as the stationary modules. You can then edit documents in workgroups (Figure 4). In the overview screen, you can even configure multiple cloud portals. Alternatively, you can edit documents in the stationary desktop version and then upload them to the cloud. OnlyOffice thus implements seamless integration of stationary and web-based services in a single interface, thus consistently avoiding any unnecessary ballast.

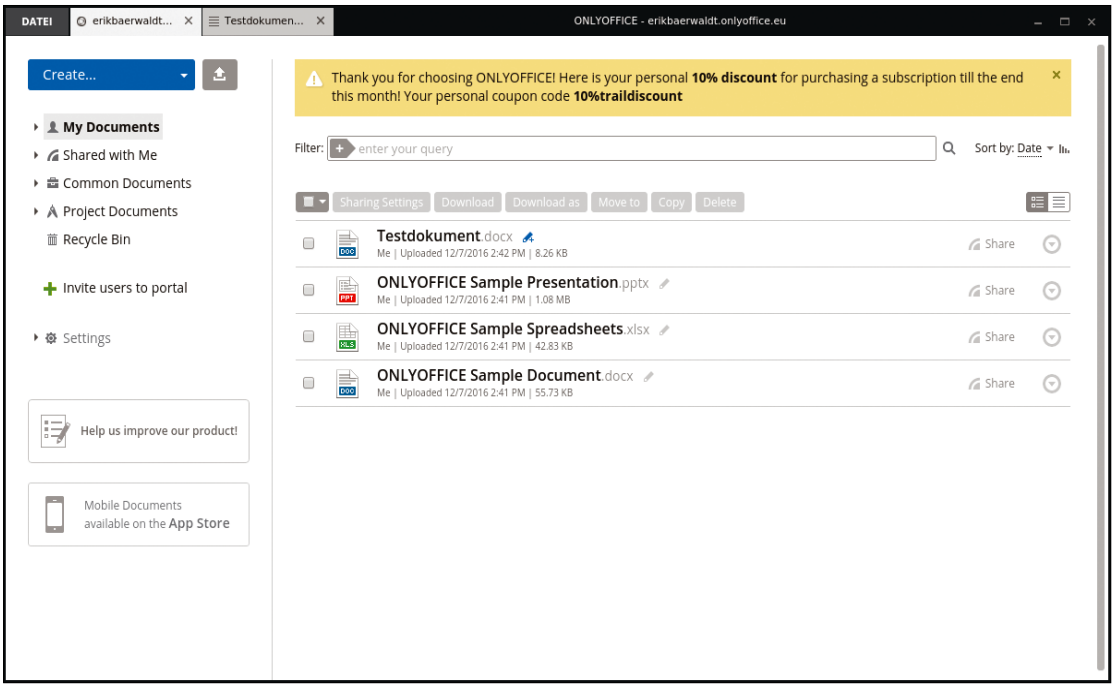


Figure 4: OnlyOffice plays to its strengths in collaborative work with multiple users.

Ascensio offers several server variants for local installation, making it possible to establish a complete work environment for group services within your own IT infrastructure. The manufacturer supports Debian and Red Hat and their derivatives via re-

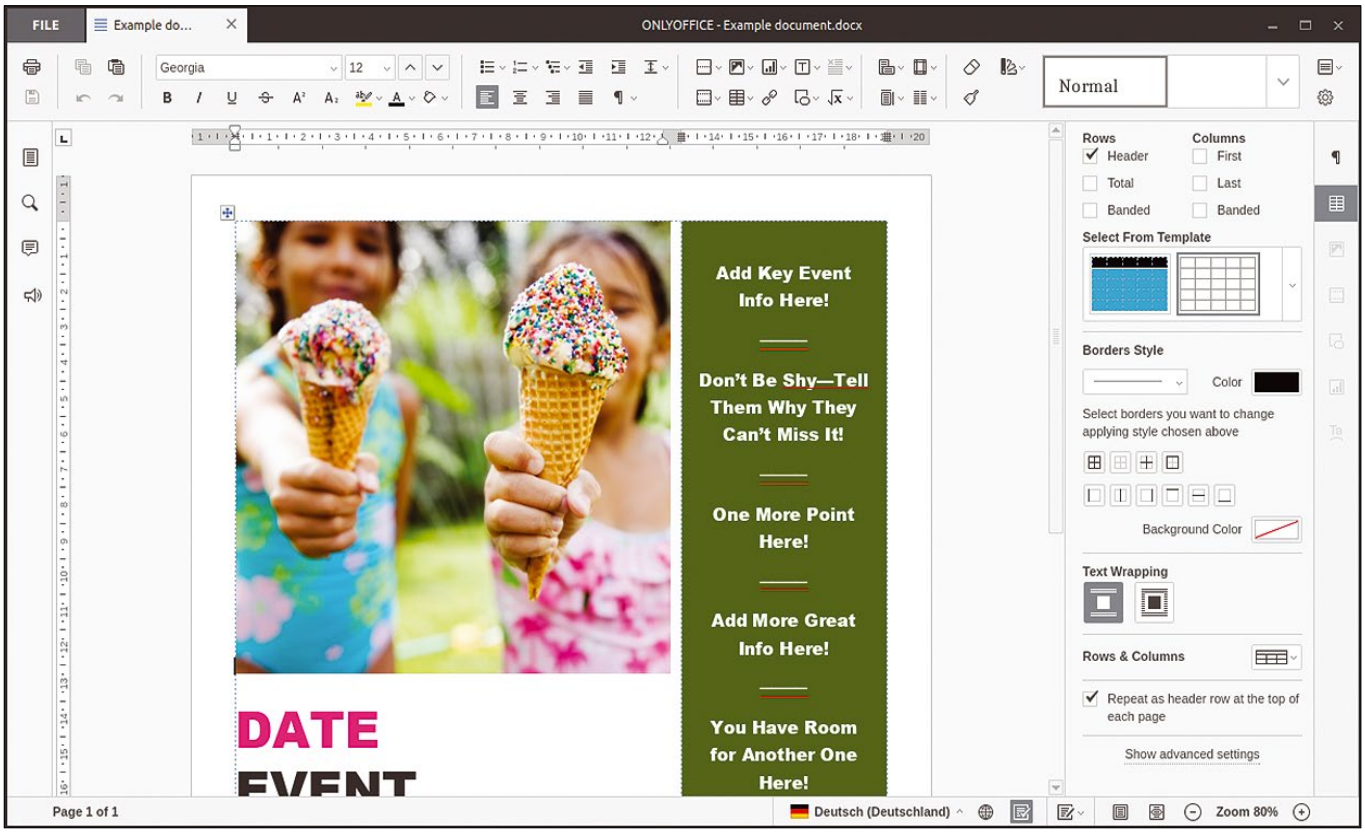


Figure 5: OnlyOffice opens documents in Microsoft formats without complaint, but it has room for improvement with ODF files.

positories [6]. You will also find Docker images. See the detailed instructions for installing the server version via separate repositories [7].

The OnlyOffice Community Server is free of charge, but a one-off payment of US$1500 per server is required for the commercial Enterprise Version after a test period of 30 days [8]. The Enterprise Edition provides a control panel and some other additional features, as well as support and updates for one year.

Foreign Language Support

Ascensio System advertises OnlyOffice as being fully Microsoft Office compatible. However, Microsoft tends to change its file formats regularly, which often compromises compatibility. Although many office suites advertise MS Office compatibility, experience shows that converting the current MS Office formats is anything but trivial, especially with complex documents. We put OnlyOffice to the test with some particularly complex `.docx` documents.

The results showed that OnlyOffice is the only office suite, apart from China's WPS Office, capable of reading documents that were originally created with MS-Office with virtually no errors. OnlyOffice also copes well with older MS Office documents in `.doc` format. The only docs that caused a problem in our tests were documents with very rarely used features such as placeholder attributes.

OnlyOffice, however, made a mess out of reading ODT files from LibreOffice: For example, it displayed boxes with input fields incorrectly, and graphic symbols partially appeared in the wrong places, making manual post-editing necessary (see Figure 5).

Conclusions

The newcomer OnlyOffice Desktop Editor does a pretty good job as a desktop office suite. The cloud connection, and the related options for efficient collaborative work, turn out to be particularly mature and practical, with the suite interlinking stationary applications and web-based work in an excellent way. Another positive feature is the good conversion filter for Microsoft document formats.

If we were to find fault at all, it would be with the modern-looking, but fairly pale, and not particularly ergonomic, user interface in the individual modules. Although the settings are uncluttered, the often-illogical positioning of the controls means a learning curve for users. But these quibbles apart, OnlyOffice provides a good alternative to existing suites and cloud services. ■■■

INFO

[1] SoftMaker Office: *http://www.softmaker.de/softmaker-office*
[2] WPS Office: *https://www.wps.com*
[3] Apache OpenOffice: *https://www.openoffice.org/*
[4] Calligra Suite: *https://www.calligra.org/*
[5] OnlyOffice: *https://www.onlyoffice.com*
[6] OnlyOffice Server Community Edition: *https://www.onlyoffice.com/de/download.aspx*
[7] "Installing Community Server for Linux on a local server": *http://helpcenter.onlyoffice.com/server/linux/community/linux-installation.aspx*
[8] OnlyOffice Enterprise Edition: *https://www.onlyoffice.com/de/enterprise-edition.aspx*

Self-organization with Getting Things Gnome

To-Do List

Getting Things Gnome helps with getting things done. Users can enter outstanding tasks and assign keywords and completion dates to their task lists in the foreground. *By Daniel Tibi*

Getting Things Done (GTD) [1] is a self-organization method developed by David Allen. The GTD approach is based on the theory that trying to maintain a complex task list in your head takes a toll on concentration. The idea is to move task tracking out of the mind and put it in the hands of a trusted external system. The system tracks the tasks so the user can focus on solving and completing the problems.

According to the GTD approach, everything you deal with – be it an article you are interested in, the tax return you need to submit, or a birthday party you are planning – passes through the GTD system and is processed according to specific criteria that help you stay organized.

The system works both with physical objects (e.g., a pen and paper, an inbox that is a real box, and a physical filing system) and with digital objects. Various applications let you implement the GTD system. One handy and free program that helps with GTD is Getting Things Gnome (GTG).

Getting Things Done

The first step is to collect all the documents that represent your various responsibilities. These documents might consist of letters, newspaper clippings, articles, photographs, and handwritten notes, as well as digital items such as email messages or even notes in a digital notebook. You work through the items that require action and complete them according to predefined criteria. The basic rule is: Anything that you have taken from the inbox must not be put back, but must be assigned to a suitable place in the system.

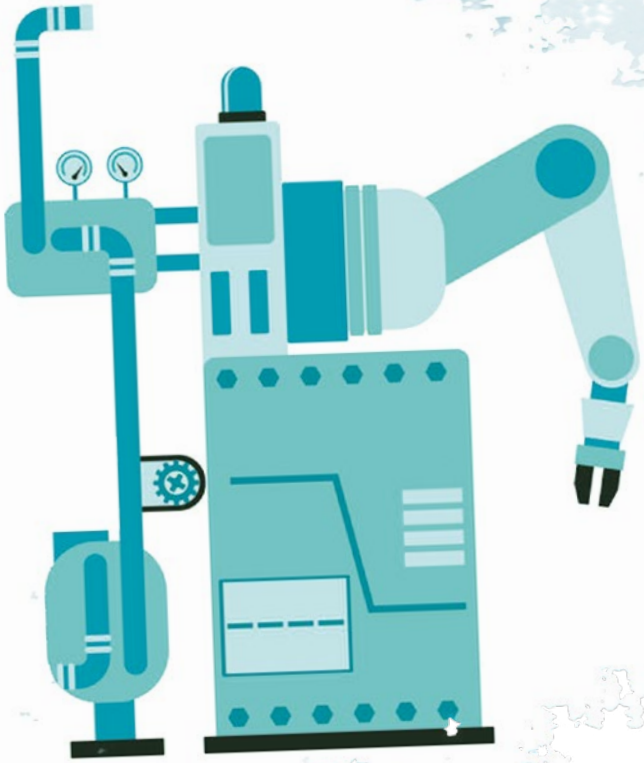
Select the items one by one. The first question for each item is: Throw it away, archive it, save it for later, or complete it? You can throw away a mail message that doesn't interest you. You can archive a newspaper article that you have read but want

INSTALLATION

Installation of the current version, GTG v0.3.1, is incredibly simple. On the project website [2], you will find both source and binary packages for various Linux systems. You can also look for GTG in your Linux distro's package repository. On Ubuntu, you can install GTG with:

```
sudo apt-get install gtg
```

Depending on the package and distro, the setup program might place a launcher in your Start menu. Otherwise, you can start GTG with the `gtg` shell command in a terminal. If you want to start the program automatically upon login, go to *Edit* | *Preferences* and check *Start getting things GNOME! on every login*.



to keep. The archive can also include things you might want to do later but don't have time for now, such as a description of a nearby yoga class. It is important to have a neat and well-maintained archive system in which you can easily find whatever you store.

Some of the tasks that you won't archive or throw away are easier to do immediately, without cluttering up the system. As a rule of thumb: Do things that take less than two minutes immediately; reserve the rest for later completion.

After separating out the things you will throw away, archive, or do immediately, you get down to something resembling a to-do list with items that require intermittent or ongoing attention. GTG is a convenient, simple tool for tracking tasks and maintaining that to-do list (see the "Installation" box). You can quickly record the task, the start date, the desired completion date, and any sub-tasks that might be required as part of the process. GTG also lets you tag tasks for easy searching by category and record notes describing the state of the task.

Creating Tasks

GTG focuses on the tasks, which the program displays in the main window. You can create a new task by clicking on the *New Task* button or using the keyboard shortcut Ctrl+N. A new dialog appears. In this example, I want to save the yoga class offer for a later date. In the first line of the new dialog, the title of the task, *Yoga Class*, appears in a larger, underlined font followed by a task description, *Penguin Yoga, penguin-yoga@example.com*.

To make it easier to find tasks, you can assign tags, which consist of an @ sign, followed by the keyword. In our example, you could assign the following tags: *@maybe/sometime @leisure @yoga*. The tags are highlighted making them immediately stand out in the text (Figure 1).

Finally, you will need to define a start and a due date. For the yoga course, you might not have time until August, and the offer is only good until October 31. There are two fields available at the bottom of the dialog: You can enter the data or select from a calendar. If there is no specific due date, just choose *sometime*. When you close the dialog, the new task will appear in the main window.

Using *View* | *Tags sidebar* or the F9 key, you can now open a sidebar in which all the tags you have assigned are displayed. Right-clicking on a keyword opens a dialog, in which you can change the color the program uses to highlight a keyword for a task. Additionally, you can assign an icon to the keyword; this icon then appears in the main window next to the task title. A search function is available in the sidebar.

Wait

You can also enter tasks delegated to others in GTG. Tracking delegation helps you keep an overview of what you assigned, for when, and to whom, and it can also remind you that completion is overdue.

For instance, you could forward a letter from the IRS, reminding you that you need to complete your income tax return, to your tax consultant. At the same time, you would use Ctrl+N to create a new task. Give the task a new title, such as *Income Tax Return: Reminder from Internal Revenue Service forwarded to tax consultant February 1, 2017.* Meaningful tags would be: *@wait @income tax*.

Your tax consultant promised to have your income tax return ready by April 1, so that sufficient time remains for you to check the results from the consultant, sign the forms, and turn in the return to the IRS by April 18. You enter this data as the start date and completion date of the new task.

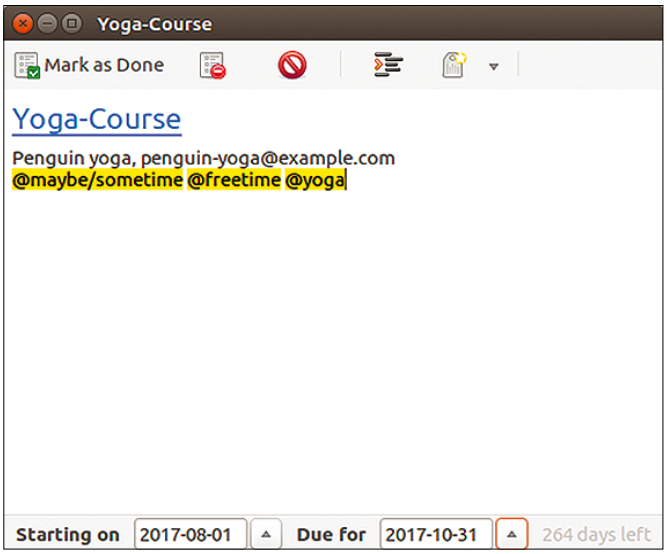


Figure 1: Creating new tasks is quick and simple. Tags help you find the tasks again.

Projects

Tasks that you want to complete yourself, and that take more than two minutes, are referred to as projects in GTD. Such projects are typically broken down into individual tasks.

First, create a new task for the overall project in the usual way by pressing Ctrl+N. If you want to write an article for a professional journal, the title could be *Write article*. The following tags seem appropriate: *@projects @article*. The completion date is the editorial deadline. Enter the individual sub-tasks line by line with a leading dash; for example: *-1. Literature research*, *-2. Get literature*, *-3. Read literature*, *-4. Write draft*, *-5. Send draft to editor*, *-6. Incorporate editorial changes*, *-7. Send finished article to editor*.

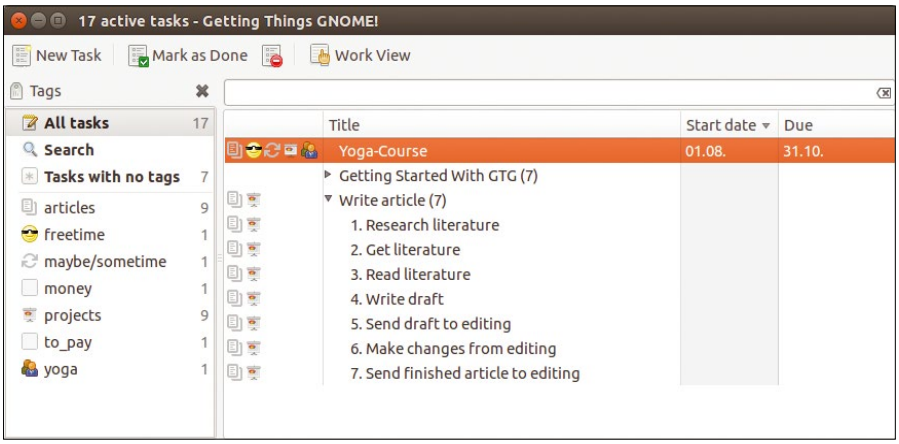


Figure 2: A plugin automatically highlights GTG tasks according to their urgency.

GTG creates its own sub-entry for each part of the task and also creates a corresponding link in the main entry's window. In the main window, the entries are shown indented under the main entry. To keep track, it is useful to enumerate the tasks in the order in which you want to complete them.

In the sub-tasks, you can save notes for each step, for example, a list of the literature you want to read. As usual, you also have the option of entering tags. After sending a draft of your article to the editor, you can to add the *@wait* tag to the corresponding sub-item.

You need to assign start dates and due dates for sub-tasks. In this way, you can keep track of which sub-task is pending; of course, the due date for a sub-task must be no later than the due date for the project. After you complete a sub-task, right click on it and select *Mark as Done* from the menu.

Regular Processing

To see which tasks are currently pending, you need to work through GTG regularly as part of your daily routine. In the main window, click the *Work View* button. Pressing the button changes the view of the main window (Figure 2). GTG only displays those tasks that you need to complete. They include tasks that either have no start date or whose start date has already been reached. If required, you can filter this list for keywords.

For example, if you are heading into town for shopping, you can display all items with the keyword *@shopping*, so that you are sure to complete all your pending purchases in a single visit. You can then highlight any completed tasks, and they will disappear from the list.

Simplification

Various plugins add functionality to GTG. To view and enable any available plugins, select *Edit | Plugins*. You'll find plugins that let you automatically delete highlighted tasks, color-code tasks for easy recognition of urgency, and more.

Use the Tomboy/Gnote plugin to sync tasks between GTG and the Gnote or Tomboy note-taking tools (Figure 3). First enable the plugin, and then enable synchronization using *Edit | Synchronization Services*. A new menu appears bottom left; now click on *Add*.

Then select the program with which you want to sync tasks and notes (for example, Tomboy) and confirm by pressing *OK*. You just need to decide whether GTG should synchronize all tasks and notes or just those that contain a specific keyword, say *@GTG-Tomboy*. Click on *Enable Synchronization* and *Close* to complete the synchronization.

Conclusions

GTD is a proven system for self-organization that helps you implement the GTD system in Linux. The lean and well-arranged program supports easy and intuitive use.

GTG is already a useful addition to your desktop toolkit, but it still has room for improvement. More integration options would be helpful, such as the ability to synchronize with other memo tools. It would also be nice to sync the start and due dates against a calendar program or an appropriate service on the Internet. Someday, it would also be helpful to synchronize tasks between different computers or mobile devices. ■■■

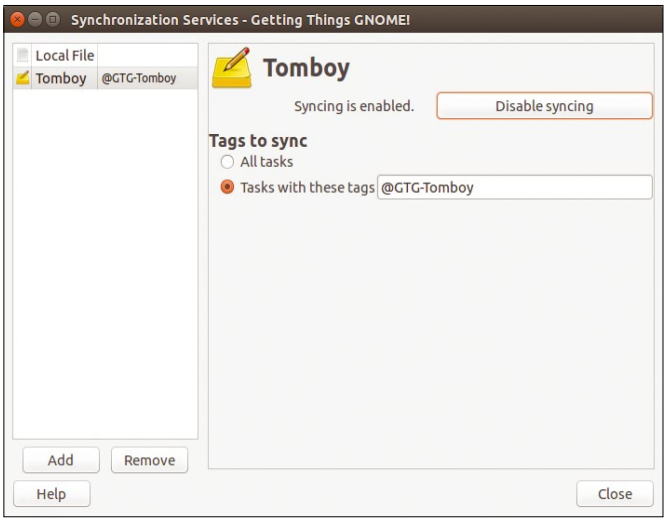


Figure 3: GTG lets you synchronize tasks with other programs, here with the notebook software Tomboy.

INFO

[1] GTD: *http://gettingthingsdone.com*

[2] GTG download: *http://gtgnome.net/download/*

Leaner surfing with the SeaMonkey Internet Suite

Clever Combination

SeaMonkey is a lean and time-tested alternative for web, mail, IM, and address management that supports Firefox plugins. *By Erik Bärwaldt*

Mozilla Firefox is regarded as one of the most secure web browsers and enjoys great popularity for its versatility and usability. But Firefox and its email counterpart Thunderbird are not really suitable for low-end hardware, and even some users with high-end hardware prefer to travel light and avoid big, over-sized applications.

Several leaner browser alternatives exist today, such as Qupzilla, Midori, Chrome, or even Min and Dillo, but many of these tools don't provide the security and privacy options included with Firefox. Perhaps more importantly, they don't offer the rich collections of add-ons and extensions available through Firefox and the Mozilla family of applications.

SeaMonkey [1] is a suite of Internet tools that provides a lean, standards-compliant alternative to Thunderbird and Firefox. The suite is based on the Mozilla Application Suite and is a descendant of the Netscape tool family. The good news for users who are searching for a leaner alternative is that SeaMonkey provides this lighter footprint while still maintaining compatibility with many Firefox add-ons.

SeaMonkey consists of a web browser, an email and news client with address book, and an HTML editor. Because the individual programs share several libraries, the suite claims significantly less space on your mass storage device and in RAM than

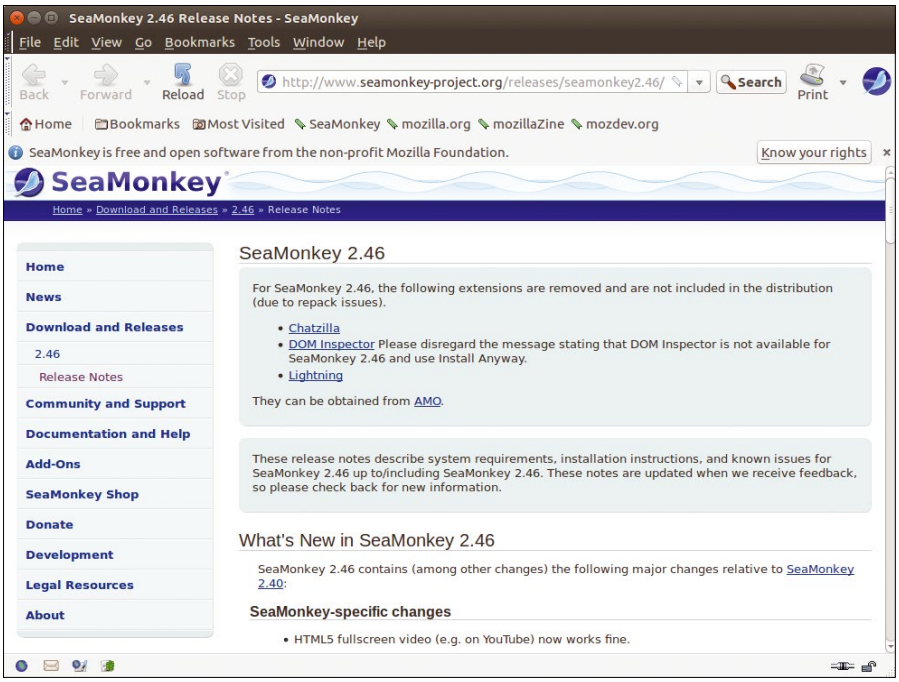
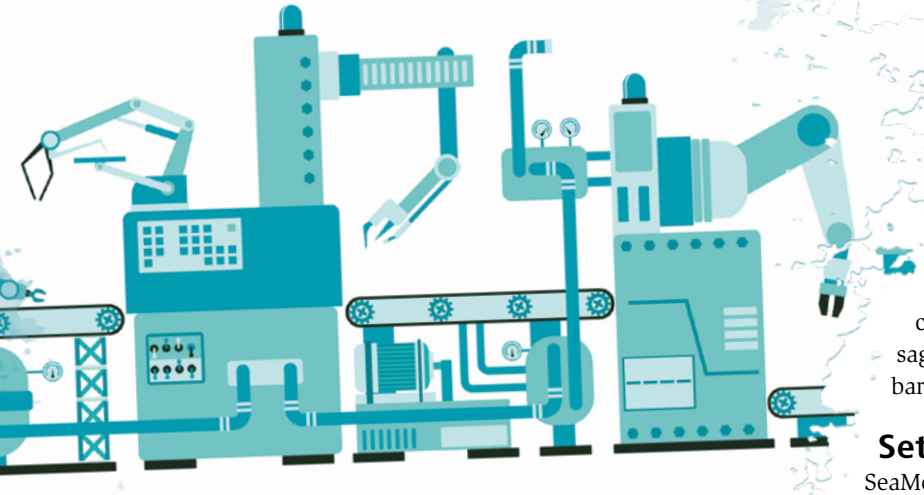


Figure 1: The SeaMonkey interface is reminiscent of the legacy Netscape browser.



Firefox and Thunderbird, both of which come with their own runtime environment. In the case of concurrent use, SeaMonkey uses half as much memory as Firefox and Thunderbird running together.

Installation

The current 2.46 version of SeaMonkey is available from the project website. The project offers the 32-bit version in various languages, but the 64-bit version is available in English only. After downloading the 46MB archive, extract it with the following command:

```
tar -xjvf seamonkey-2.46.tar.bz2
```

Then change to the newly created program directory and launch the software with the `./seamonkey` command. On 64-bit systems, the 32-bit version of SeaMonkey sometimes fails to launch despite correctly installed libraries – in that case, you will need to use the 64-bit version.

After you launch SeaMonkey, you first need to select the checkbox to define the tasks for which SeaMonkey will be your standard application. You are then taken to the web browser's interface, which is still faintly reminiscent of the old Netscape. The interface includes many settings that have been completely deleted, or at least modified, in current browser versions (Figure 1).

SeaMonkey offers a conventional menu bar, as well as a slightly rustic-looking toolbar with a URL input field. In the default setting, the window integrates a horizontal bookmarks bar. Using the small buttons on the left of the window, you can hide or show the bars to avoid losing sight of required content and increase the space available for the display.

The status bar at the bottom of the window displays the requested URL and allows rapid switching to the composer, address book, and email client via corresponding buttons. On the right, messages from your installed add-ons are displayed; a small bar graph visualizes the load progress for websites.

Settings

SeaMonkey provides a conventional *Preferences* menu, as well as giving you the ability to manually modify individual options. Open the *Preferences* menu by selecting *Edit* | *Preferences*. The menu contains dialogs sorted by application that let you customize the program suite. Settings that apply to all components alike are grouped in *Appearance*, *Privacy & Security*, and *Advanced* (Figure 2).

You can use the menu bar to access several dialogs that allow rapid changing of specific configuration options. For example, the *Tools* section contains a considerable number of managers, each of which handles a specific task: The Cookie Manager lets manage the cookie options; the Graphics Manager helps you tune the graphics settings, and the Popup Manager lets you open popup windows that the browser blocks by default.

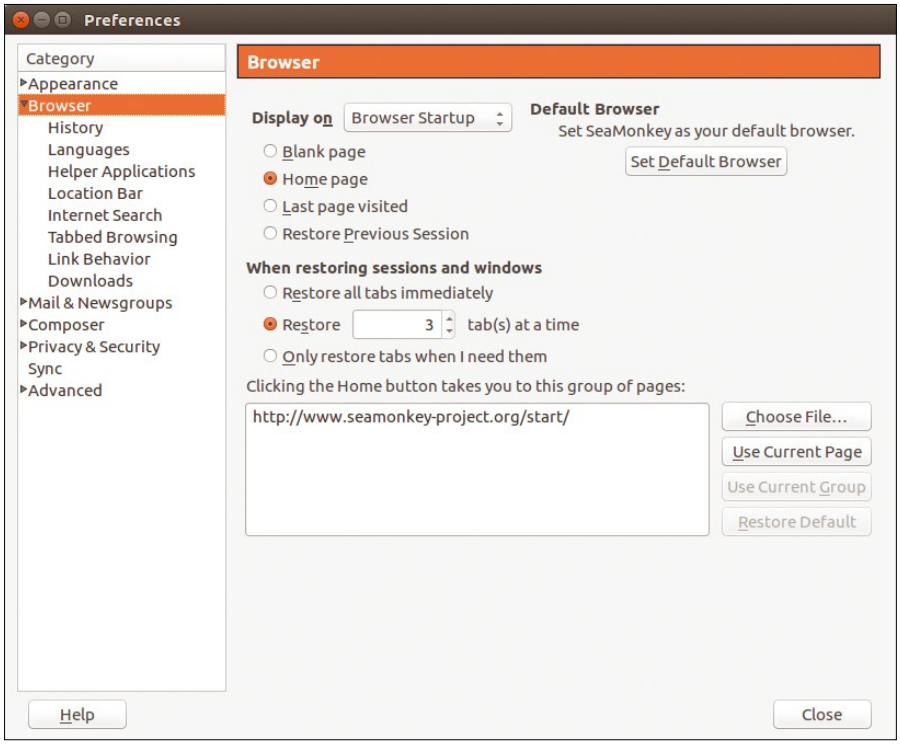


Figure 2: SeaMonkey combines the configuration options for the various modules in a single window.

Other managers take care of downloads and passwords. In the same menu, you will also find the entry *Clear private data…* , which lets you to remove histories, cookies, and other data accumulations. The *Change profile…* menu item lets you enable and manage additional user profiles in a separate dialog.

Design and Function

SeaMonkey's roots date back to the beginnings of Netscape Communicator in 1997, when Netscape was still the leading suite of web applications. SeaMonkey thus has a slightly old-fashioned graphical interface by today's standards, but one that supports intuitive use with many keyboard shortcuts.

Although the Gecko rendering engine provides good support for HTML5 under the hood, the interface does cause some irritation at first glance: You will initially look in vain for today's typical tab structure with individual tabs for the requested web pages; most browsers have dropped the button and menu bar – still implemented by SeaMonkey – for a 3D look with larger buttons.

But instead of restricting the configuration options for users to just a few switches, SeaMonkey takes a different approach: The settings dialogs let you customize the software in detail to suit your own needs. For example, in *Browser | Tabbed browsing*, you can uncheck the box for *Hide tab bar, if only one tab is displayed* to permanently show the tab bar.

You can optionally use the keyboard to manage the tabs: For example, Ctrl+T opens a new tab, and Ctrl+W closes the current tab. If multiple tabs are open, matching icons appear on the left and right in the bar for opening and closing the tabs.

The easiest way to use the zoom function is with the keyboard shortcut Ctrl++ (larger) and Ctrl+- (smaller). Alternatively, you can access this feature in the menu below *View | Zoom (100%)*, which provides several preset zoom levels. To toggle between the different applications, you can either use the *Window* menu or the shortcuts Ctrl+1 to 4. The original program is always left open.

For example, to launch the email client directly from inside your browser, you can press Ctrl + 2; Ctrl + 5 launches the address book. These keyboard shortcuts are available in all SeaMonkey applications. A download manager in its own window provides an overview of the current status at all times. A search function lets you track specific files or archives (Figure 3) in long download lists.

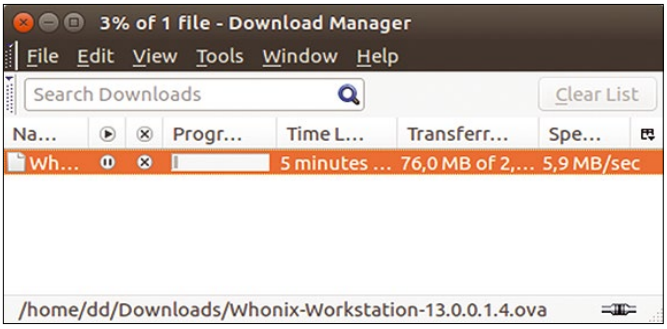


Figure 3: The integrated Download Manager provides a detailed summary of current and past downloads and comes with a search function.

Applications

The interfaces of the SeaMonkey applications are based on the appearance of the browser and thus require practically no training. The built-in email client supports both POP3 and IMAP. When you launch the mail client, a wizard guides you through the configuration (Figure 4). Even the address book built into SeaMonkey comes with a wizard.

The Composer provides an interface for designing your own web pages, removing the need to create the basic formatting by manually inserting HTML tags into the source code itself; just a few mouse clicks on a toolbar are all it takes. The program is thus more reminiscent of a word processor and can be used productively after quite a short learning curve.

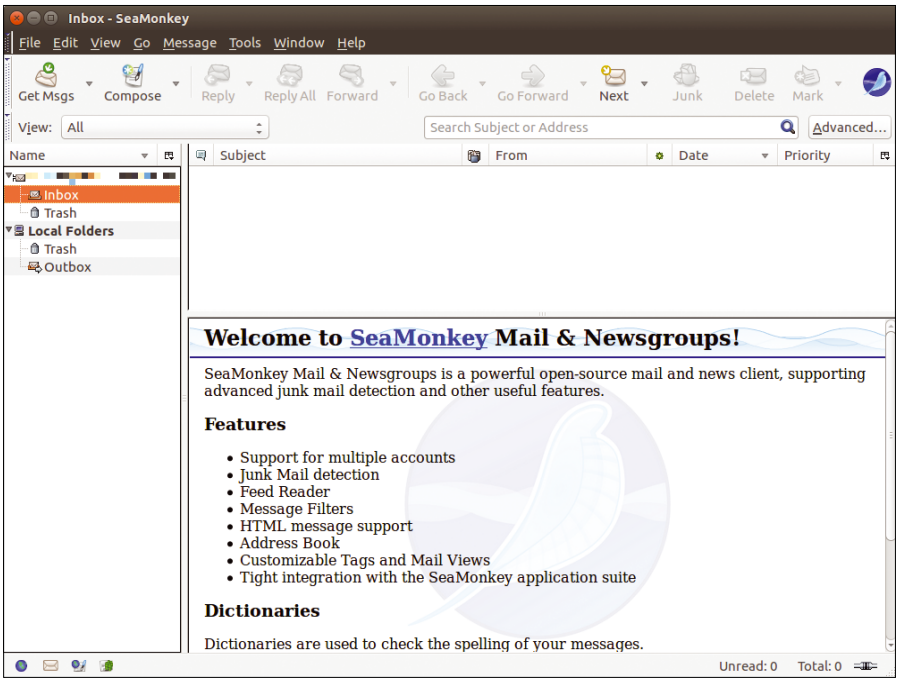


Figure 4: The mail client is designed for intuitive control and has a structure oriented on Thunderbird.

Extensions

You can extend SeaMonkey with many of the add-ons that are available for Firefox and Thunderbird. The project has its own add-on page for this purpose. But because SeaMonkey has a far smaller community, the number of extensions remains manageable.

Not all of the Firefox add-ons are officially supported by SeaMonkey, but in our lab, we discovered that even some of those that aren't officially supported will still work. You will often need to manually download the file from the Mozilla site and then set it up in the application suite using *File | Open file* dialog.

You can spice up SeaMonkey's slightly rustic appearance with the selection of Firefox themes on the Mozilla page: Virtually all of the compilations for Firefox are also suitable for SeaMonkey. An enabled theme is used with all program modules. To manage themes and add-

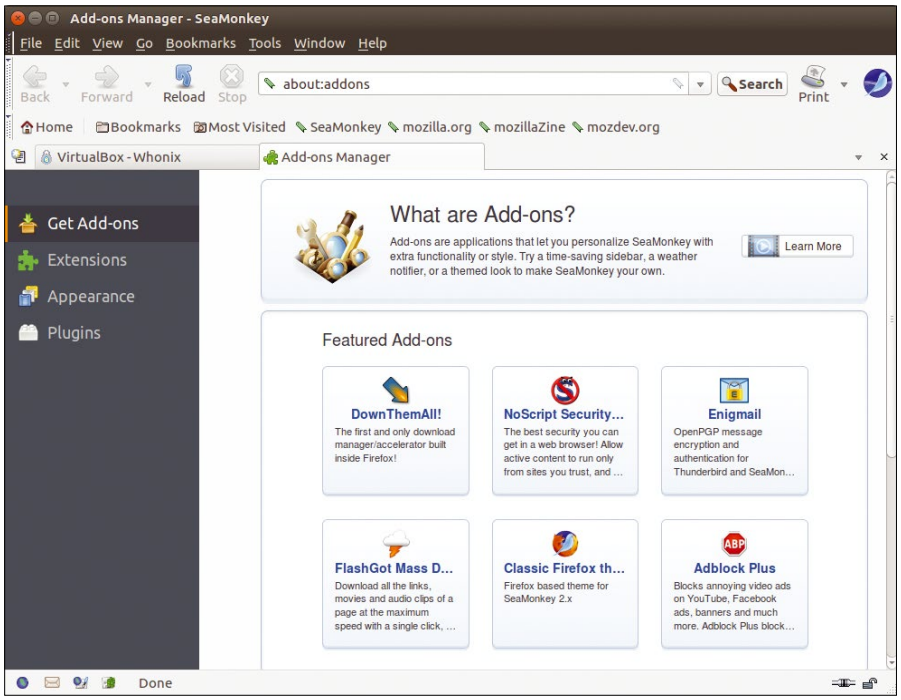


Figure 5: The SeaMonkey Add-ons Manager looks much like its Firefox counterpart.

ons, open the menu below *Tools* | *Addon Manager* (Figure 5). You can also use Firefox Bookmarks and bars in SeaMonkey. Because both programs use the same file format for the bookmark list, you just need to save your bookmarks in the Firefox bookmark manager and then import them into SeaMonkey.

Security

Thanks to its robust Mozilla underpinnings, the SeaMonkey browser is one the securest alternatives to Firefox. SeaMonkey does not collect any user-specific information, such as telemetry data and status reports. Only a crash detector is implemented to gather information for improving the software, but you can switch it off.

Important add-ons like Adblock Plus and YesScript can be easily integrate with the suite using the standard dialog. You can also download a slightly older version of the Ghostery anti-tracking tool from the Mozilla site and install it in SeaMonkey. Even the uBlock Origin tracking and advertising blocker is suitable for SeaMonkey, but you do need to set up it manually.

For an overview of the currently compatible add-ons, check out the list by the developers at [2]. In addition, you also have the option of converting locally downloaded add-on files to the SeaMonkey format on a web page [3]

Hand-Crafted

Much like Firefox, SeaMonkey offers an advanced configuration dialog, which you access by typing *about:config* in the address bar of the browser. After confirming the warning prompt, you are taken to a page with configuration settings in a list (Figure 6). To find specific parameters, enter the desired term in the search bar. A click on the relevant entry opens it. However, you should exercise caution, because incorrect settings can break the browser.

Conclusions

SeaMonkey is a full-service web suite for users who are looking for a lean alternative to Firefox and Thunderbird. Thanks to the common underpinnings, SeaMonkey keeps pace with Firefox in terms of security. Integrating some add-ons might require a little manual work to convert Firefox add-ons to a format supported by SeaMonkey.

The 64-bit version is only available in English as of this writing, but a clear menu structure and simple nomenclature make the English version accessible for those who use English as a second language. SeaMonkey is extremely stable and fast, and its support for current standards such as HTML5 mean that it's more than ready for production use. ■■■

INFO

[1] SeaMonkey: *http://www.seamonkey-project.org/*

[2] Add-on compatibility list: *http://addonconverter.fotokraina.com/compatibility/*

[3] Add-on converter: *http://addonconverter.fotokraina.com/*

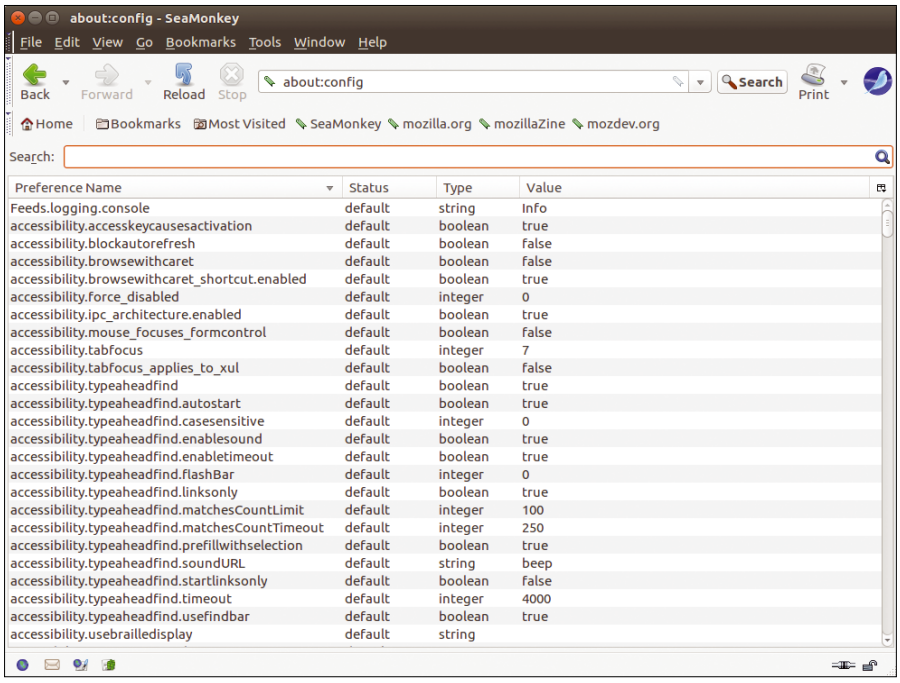


Figure 6: Like Firefox, SeaMonkey gives users access to settings hidden deep in the system via about:config.

Graphical desktop front ends for KVM and Qemu

ZOO KEEPER

If you want to care for a zoo full of exotic KVM guest systems on your desktop, you could use a little help from a graphical front end. *By Erik Bärwaldt*

Virtual work environments exploit the power of modern multicore processors by efficiently virtualizing complete operating systems and their services. The kernel-integrated Kernel-based Virtual Machine (KVM) [1] is a useful alternative to partly proprietary virtualization software from Oracle (VirtualBox [2]) or VMware (VMware Workstation [3]).

The matching kernel module has been around since Linux 2.6.20, and Red Hat has managed it since 2008.

The advantage of KVM compared to the third-party solutions by Oracle and VMware is its high execution speed and resource-friendly work approach. One disadvantage is the somewhat complex configuration of these kernel-based virtual machines.

NOT RATED

Miscellaneous desktop GUIs for the KVM and QEMU duo abound in the Linux universe. Many of them, including GKVM [6], Qemulaunch [7], or Qemulator are no longer developed and maintained, so that the applications often do not run on current distributions, or at least only with incommensurate effort.

QtEmu [8] is not well maintained; the packages for several distributions have included unresolved bugs for years.

Other programs such as Qemu Manager [9], which emerged from Qemulaunch, is not localized in English and is thus difficult for users in the English-speaking world to come to grips with. The very young, but interesting Kimchi [10] is currently difficult to install on most distributions and is also still very much under active development. However, *Linux Magazine* has tested most of these programs, including the somewhat ancient live distribution VirtLive [11].

At the end of the day, we only picked the programs that we were able to install without any problems using the test distributions Linux Mint 18 (Sarah), Open Mandriva LX 3.0, Ubuntu 16.04 LTS (Xenial Xerus), Rosa Desktop R8, and openSUSE Tumbleweed, and which provide reliable service there.

A number of graphical tools help reduce the administration costs for compute clusters and servers that manage multiple virtual guests. Examples include oVirt [4] and Proxmox VE [5]. A web interface, which requires some serious configuration work on the part of the admin, is often used as a GUI. But industrial-strength tools like oVirt and Proxmox VE are overkill for simple virtual desktop environments.

A smaller and lighter class of graphical tools help users manage KVM systems on the desktop. Some of these tools receive very poor maintenance and thus are not really viable for Linux production environments (see the box “Not Rated”). But a few of the more well-maintained virtualization managers for KVM offer some significant benefits: Thanks to their low latency and high execution speed, the guests managed with these tools perform well for time-critical applications. And because virtual machines can emulate different processor architectures, developers can develop their programs on a single platform and then test them for many different architectures.

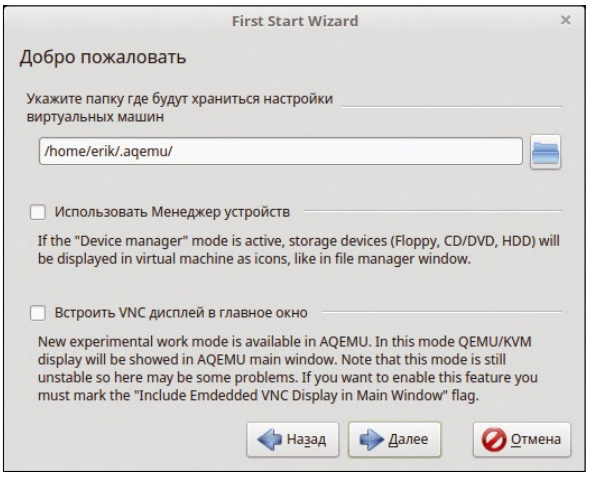


Figure 1: Aqemu: German localization is not complete.

Complete operating systems can be tested directly from the ISO image and without an intermediate step involving external mass storage. Adventurous users or admins can now test new operating system versions with little effort using the KVM system.

Requirements

KVM requires a processor-implemented hardware virtualization. The following command on Linux tells you if your system supports this feature:

```
grep -E "vmx|svm" /proc/cpuinfo
```

One of the two flags should appear in the output. If both are missing, you may only need to enable hardware virtualization in the BIOS. Using `lsmod | grep "kvm"` shows whether the system loads the kernel modules correctly. Depending on the processor the results should include the two lines `kvm` and `kvm_intel` (for Intel VT) or `kvm_amd` (for AMD-V).

As a further precondition, you need to install the Qemu virtualization environment [12] as KVM is used only as an interface of the operating system kernel. Qemu is available in the software archives of virtually all leading Linux distributions; users can easily install it with the respective package management tools.

Additionally the QXL graphics driver and the Spice Protocol [13] should be running on the host system because the team of KVM and Qemu only emulates an ancient Cirrus video card by default. This emulation and its VGA counterpart do not support a contemporary screen resolution of the virtual system. Spice also transfers audio data, allowing users to view movies in the virtual machines.

Integrating these into the Linux system via the command line not only proves cumbersome due to the extensive command syntax, but it is also a disadvantage if you often use different virtual systems.

At this point, the graphical KVM management tools enter the game.

Aqemu

Aqemu [14] based on the Qt libraries is visually strongly geared to the VirtualBox environment from Oracle. Users can use Aqemu out the box without training. However, an installation wizard appears when first called to prompt you for your choice of locale (Figure 1).

Then the program window appears (Figure 2). It is divided into three parts: At the top, there is a small horizontal menubar and below it a buttonbar with commonly used functions. On the left, a vertical list area enumerates the installed virtual guest systems. To its right, like in VirtualBox, there is an area that displays the settings for the individual virtual machines in seven tabs.

Pressing the green + button at the top left lets you define a new virtual machine; an ISO image can serve as a template. After assigning a name, you can move on to configure very detailed settings for the virtualized hardware. This not only relates to the processor type and the hardware architecture, but also the memory, the emulated sound card, the drives, and the network connectivity. You can also set up serial, parallel, and USB interfaces here. Advanced users will quickly understand the options as they are similar to those in VirtualBox (Figure 3) in terms of appearance.

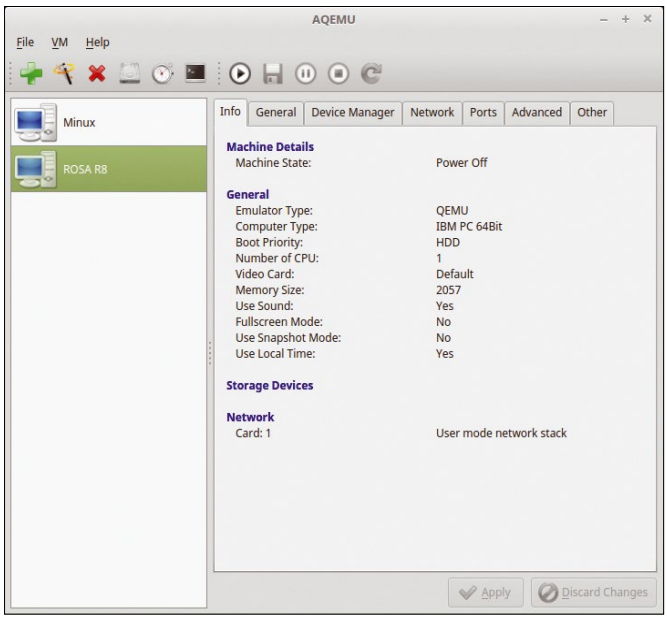


Figure 2: The splash window in Aqemu seems rather spartan, but it is also uncluttered.

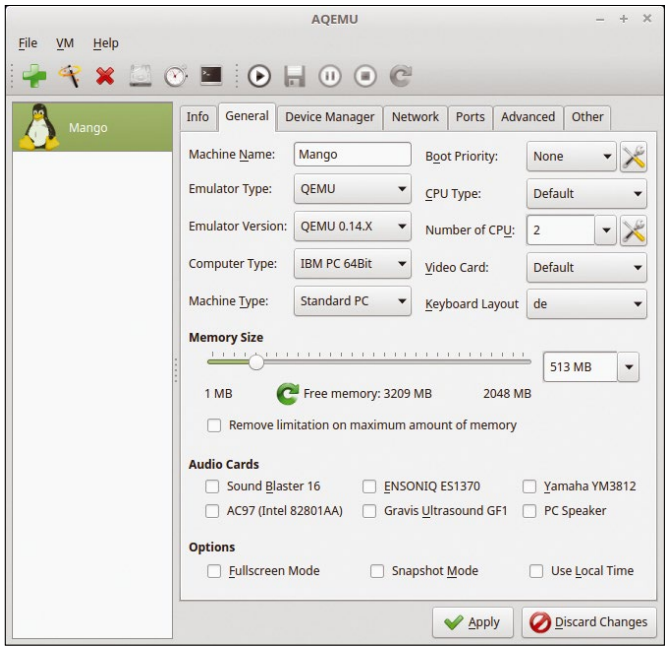


Figure 3: The configuration options for the virtual machine in Aqemu are very extensive. This might seem familiar to VirtualBox users.

Beginners can turn to the Wizard for help. You can launch the Wizard via the buttonbar; it then guides you with a couple of dialog boxes to a new virtual machine, by automatically assigning many basic settings. However, some manual work is required after the automatic configuration. The Wizard does not set up your keyboard layout, and it does not modify the boot order. This can mean that the virtual machine is not fully operational.

In the *General* tab of the Settings menu, look for *Boot priority*; check that the boot media are in the correct order and check the device list in the *Device Manager* tab. You may need to modify these. The routine always creates a virtual hard disk here, but does not ask for the path to it when booting an ISO image. You can add this manually to the device list: *Add CD/DVD-ROM* (Figure 4).

Once the configuration is complete, press the Start button (with the small triangle) to boot it. This will boot the virtual system in a separate, automatically scaling window; a scaled-down preview function as offered by VirtualBox is not available.

In a direct comparison with Oracle's software, the sheer pace at which the KVM machine runs is striking. Regardless of the size of the virtual machine and its desktop environment the KVM-based Aqemu achieved a speed that nearly matched that of a dedicated system in our lab.

One special feature in Aqemu is the option to adjust the hardware architecture with a mouse click. Below *General* | *Computer Type*, Aqemu offers a large number of supported hardware architectures, ranging from ARM-based systems, through Power PC and Sparc computers to 64-bit Intel PCs.

Sometimes virtual machines require you to tweak individual settings to achieve superior performance with the virtual environment. This applies in particular to the RAM and CPU usage presets, which the Setup Wizard rarely sets up adequately. Aqemu's very detailed options make it possible to test the run-time behavior of an operating system in a virtual machine with different hardware emulations. The software even emulates an Intel 80486 processor and supports older sound and video cards, which are suitable for experiments.

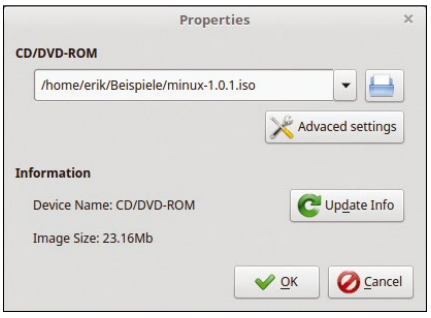


Figure 4: Booting from an ISO image is no problem for Aqemu as long as the user specifies the path manually.

If required, users can define alternative images on external disks, to start a virtual machine. Thanks to the support for serial and parallel ports, you can still use older printer models with the IEEE 1284 interface. The trio of KVM, Qemu, and Aqemu is thus especially recommended as a solid platform in heterogeneous infrastructures.

Virtual Machine Manager

The Virtual Machine Manager (VMM) [15] acts as a graphical add-on for KVM like Aqemu, but also supports the Xen virtualization software, as well as LXC containers. The software can also be found in the repositories of most major distributions. In addition to the option for GTK+-based interfaces, there is also a version for Qt work environments.

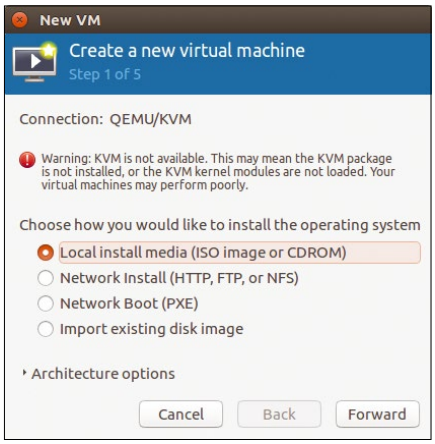


Figure 5: VMM makes it easy for users to create a new virtual machine, as a wizard provides support.

After completing the installation, you will find a corresponding entry in the System Management menu that takes you to a very spartan interface. As VMM requires multiple prerequisites for correct operation, the software first checks at startup time whether all requirements are met and indicates missing packages or maybe that the virtualization daemon has not yet been launched. If there are no warnings, the software is ready for use.

First, use *File* | *New virtual machine* to create a virtualized system. A wizard helps you do this in five steps (Figure 5).

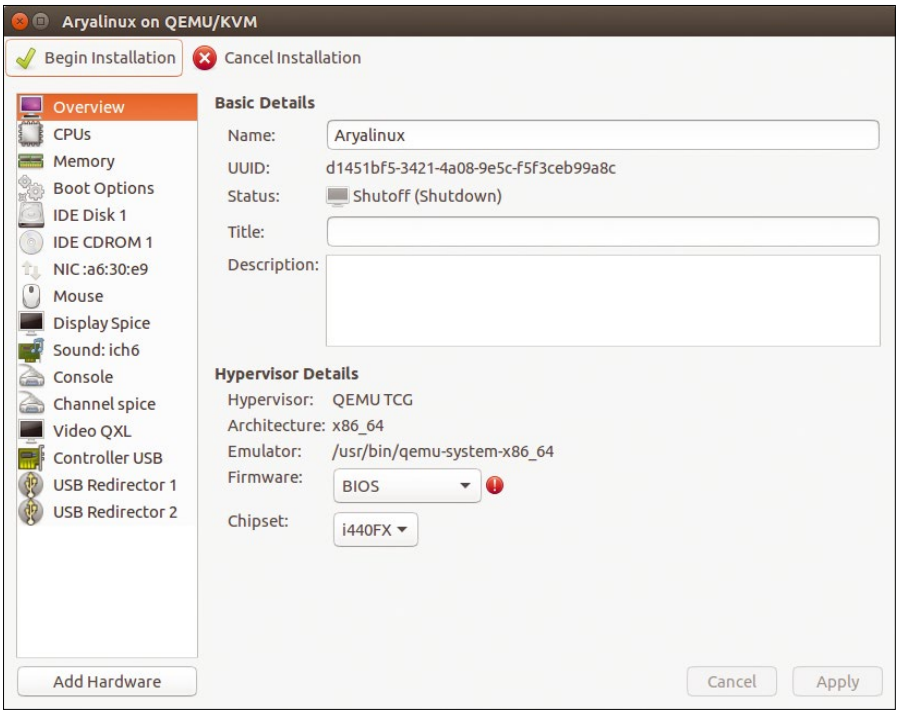


Figure 6: In VMM, too, the user can adjust the virtual system if desired.

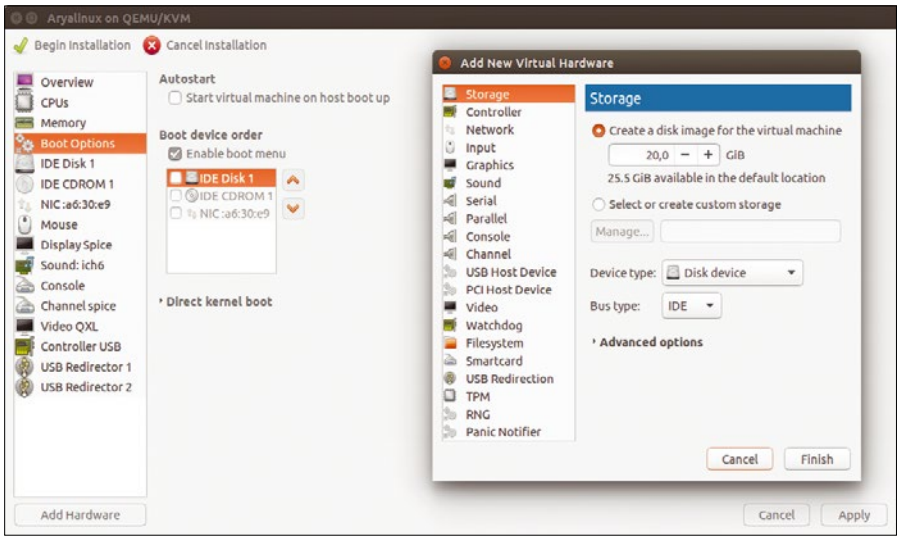


Figure 7: In VMM, integrating an ISO image is a somewhat complicated process.

In VMM, like in the other two graphical front ends, it first asks you for the source media, the RAM size, and the number of processor cores. It also sets up a virtual hard disk with a variable selectable size. If you have created drives with other virtual environments, you can use these with VMM.

To fine-tune the virtual system, your best bet is to check the *Edit configuration prior to installing* option. Clicking on *Finished* bottom right takes you to a clear-cut configuration dialog similar to that of VirtualBox (Figure 6).

At this point, you can set up the emulated hardware of the virtual guest system in great detail, without requiring additional packages and extensions. When you are done, click *Start installation* top left. After creating the guest machine, it immediately launches in a separate window.

The main screen displays the list of virtual guests and graphically highlights active VMs. A load chart on the right in the window clearly shows whether a specific guest system is busy. A permanent full load is also indicative of incorrect settings.

You can control the virtual machine through the horizontal buttonbar at the top of the program window. Pressing the icon on the left creates a new guest system. VMM provides additional statistical data on the host and the guest system: If you want to adjust the configuration as the performance is slow, you first need to enable a host system load display in the *View* | *Chart* menu. A column *Host CPU load* on the right in the main window displays CPU usage in near real time.

To change a configuration, you can tweak the many options accessible via *Open* | *Views* | *Details*. On the left, a vertical list view displays the hardware components of the system. If you click on it, you are shown the associated options on the right in each case. You can use this to retroactively tune, for example, the memory size and the processor performance.

Selecting *Boot options* solves a fairly annoying phenomenon: If you opt for an ISO image as a bootable disk, VMM does not use the ISO image as the boot medium after a reboot, but attempts to start from the empty virtual hard disk. To resolve this error, you need to press the *Add device* button bottom left and add a new second IDE-CD-ROM-2 drive. The CD-ROM device that you need to create in the window that appears adds a *Select or create custom storage* option to the guest system.

You also need to explicitly enable the new drive as the first boot medium and confirm your choice by pressing *Apply* at the bottom right (Figure 7). The virtual machine now starts from the ISO image.

VMM enables simultaneous operation of several guest systems with limited hardware resources. The load indicators are then displayed in parallel. Selecting *Show details of the virtual appliances*, at top left on the screen of the active virtual machine, and *Performance* shows you the CPU load, RAM usage, and disk and network throughput in a graphical display. The displays are useful for troubleshooting (Figure 8).

Jqemu

The Jqemu [16] software is based on Java, as the name suggests. Your advantage: It works across platforms and offers the same interface everywhere. Jqemu is not too picky on Linux and can use either Oracle's Java environment or the free OpenJDK runtime.

The application is available from the website as a prebuilt (32- and 64-bit) package and can be used immediately after unpacking the zip archive. To do so, launch a terminal by typing:

```
java -jar jqemu.jar
```

You do not need administrative privileges for this. If you want to add Jqemu to the menu structure of the host operating system, you will need to do this manually.

After starting, Jqemu shows the user a clearly arranged window with three areas: Various settings and options are available in the top horizontal menubar; the buttonbar below takes you to the most important functions. The list window on the left displays one or more virtual machines.

A large configuration area waits for the virtual machines in the right pane of the program window. Multiple horizontal tabs divide the area into groups. The settings are initially modeled on those of Qemu: Jqemu suggests assigning a no longer quite state-of-art 128MB RAM to each newly created virtual machine – this is not enough for most virtual systems. The

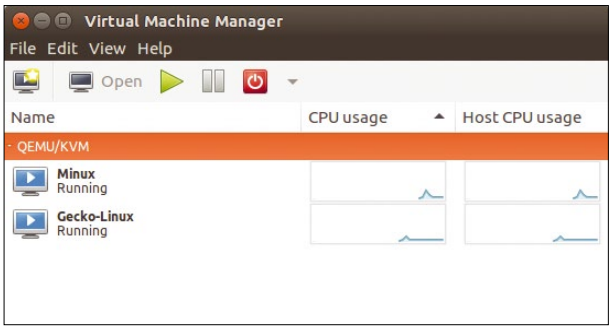


Figure 8: Two virtual desktops running simultaneously on a Linux Mint host and displaying their CPU and RAM usage.

other options need some attention: For 64-bit and 64-bit ISOs, you must select the correct Qemu binary file as Jqemu assumes the 32-bit version by default.

Jqemu provides similarly extensive options to the other subjects when it comes to configuring the guest systems. But Jqemu arranges them in a more confusing way than Aqemu or VMM.

Because the software does not have a help function, some settings cause confusion: For example, the *Network* gives you the ability to configure network access. This is not necessary in some cases, as the guest systems automatically set this up and Internet access is available without further intervention.

The graphics and audio settings need special attention. An incorrect graphics setup or faulty audio options can cause the virtual machine to freeze during startup. Some documentation would be useful here, particularly for newcomers (Figure 9).

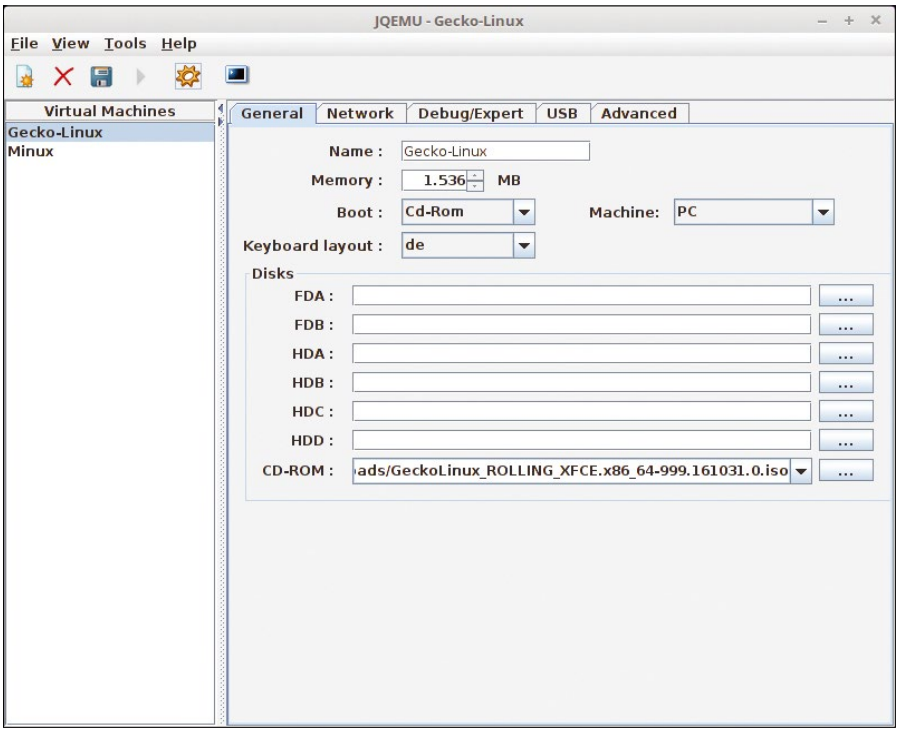


Figure 9: The configuration settings for the available drives require special attention in Jqemu.

To use an ISO image as a bootable disk, the Jqemu user has to modify several options. In the *General* tab, look for the *Boot* entry and enable the *CD-ROM* option. Then, in the same dialog, enter the path to the ISO image for the *CD-ROM* entry below the *Disks* section.

Increasing the memory size in *Memory* is also advised. Although Slim Linux distributions like Minux or Slitaz will run easily with the default 128MB, all of the well-known Linux systems need at least 1GB memory in the virtual machine. You will also want to go to the *Advanced* tab and increase the number of processor cores in the *Simulate an SMP system* option, as the larger distributions often run faster on multiple-core systems.

The Options menu, which you can access via the buttonbar, defines the supported processor architectures. In the *QEMU PC* field of the *Dependencies* section, you can define the path to the Qemu binary files, including the actual emulation file. The Qemu files typically reside in the `/usr/bin/` directory. You can then select the required emulation from a small file manager (Figure 10).

If you do not just want to run live systems operate on your virtual machines, but actually install the systems, you need to enter the path information for the drives to use in *General* | *Discs*. Unlike Aqemu, Jqemu does not automatically create the disks, but expects you to point the way here. You can use the *Tools* | *Images* | *Create* menu to create a disc image of any size, selecting from a large number of image formats as you do so. This will ensure that images work with other software packages. Jqemu is the only single application that also supports the VDI format, which Oracle's VirtualBox uses as the default format for virtual machines.

If the hardware supports this, Jqemu also allows the simultaneous operation of multiple virtual machines on a host. It packages them – like the other candidates – in dedicated instances of Qemu. They are thus completely isolated from each other. In contrast to other solutions, however, the admin has to manually shut down the individual virtual machines in their respective windows on exiting – centralized switching via ACPI commands, such as in VirtualBox, is missing.

The user does not see an overview of the resource consumption of open machines, unlike with VMM. Jqemu is thus less of a central control instrument for virtual machines and merely a graphical front end and configuration interface for Qemu.

Jqemu also provides a log function that helps you find and resolve errors in the configuration of a virtual machine. But you first need to enable logging manually. To do so, go to the *Debug/Expert* tab and enable the option *Output log items to /tmp/qemu.log*. Using the *Log Items* option, you can then define the content you want to log, to keep the logfiles understandable.

In the *Kernel Image* and *Kernel command line* input lines, you can also select a special kernel image in case of problems or pass some parameters in to the kernel at startup time, to ensure reliable operation even of more exotic systems.

In Practice

In practical use on various Linux distributions, the three graphical front ends showed no compatibility issues with a team made up of KVM and QEMU.

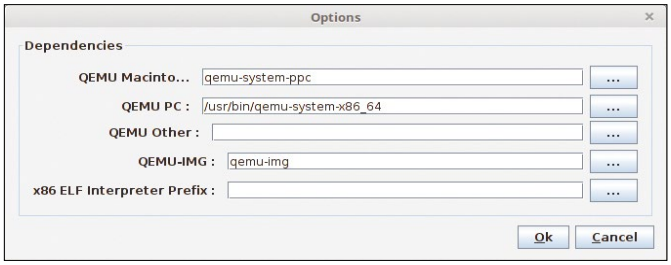


Figure 10: You can choose the hardware architecture manually in Jqemu.

However, the working speed varied enormously: With Jqemu, significant effort was required to run the operating system at a reasonable speed on multiple Linux guest systems. The Java application was also the slowest to launch.

Aqemu impressed as the most complete candidate despite very extensive setting options: The program supports virtually all virtual systems without time-consuming manual work and does so without any noticeable loss of speed. Additionally, the GUI – which visually strongly resembles Oracle's VirtualBox – was the easiest to use.

Both Aqemu and VMM impressed with their efficient use of the host computer's resources: For example, we were able to simultaneously run multiple virtual machines with just 4GB RAM and 4 Core i5 processors in our lab without revealing any bottlenecks in terms of resource allocation.

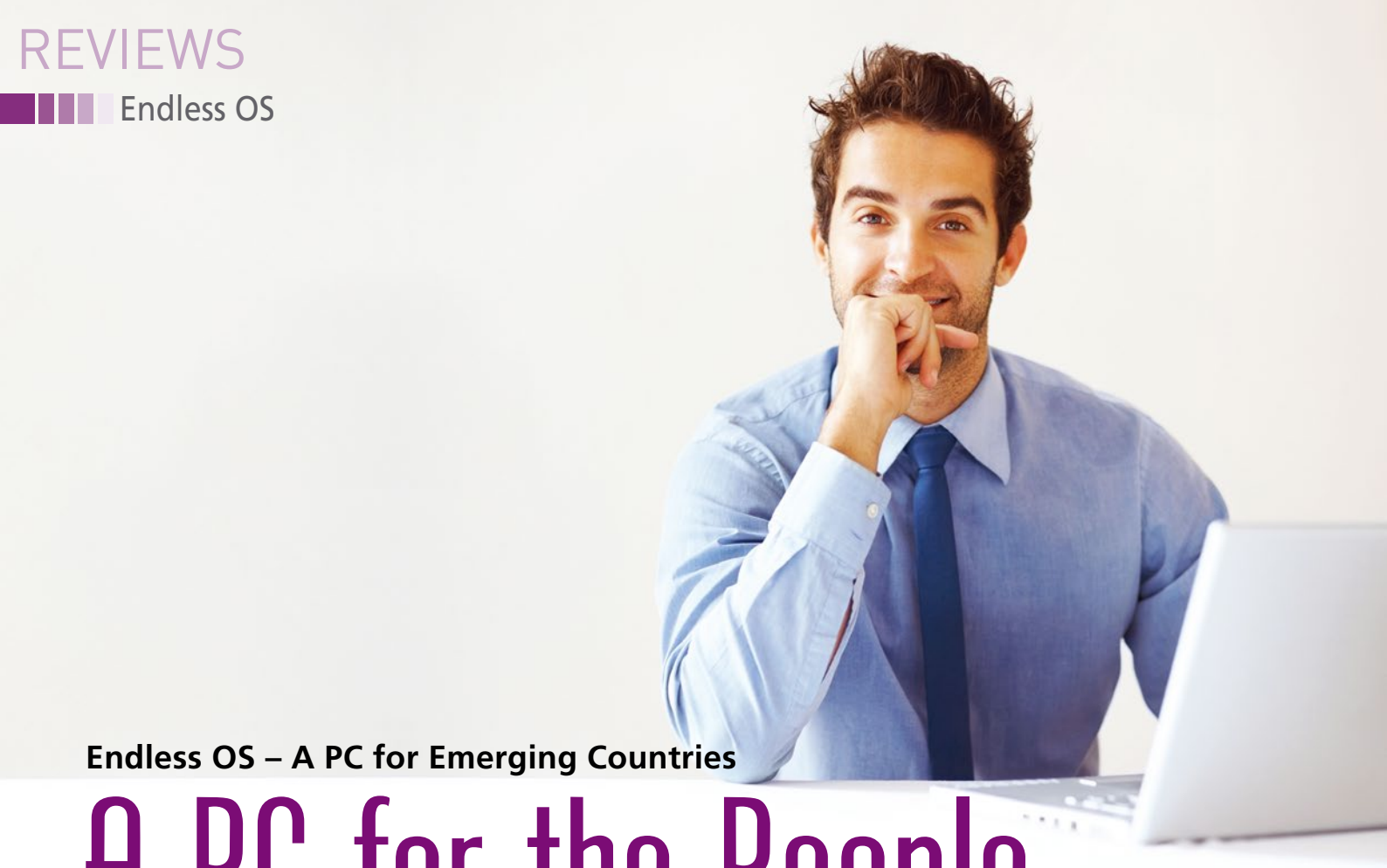
Conclusions

As a dynamic duo, KVM and Qemu with a graphical front end offer various benefits in comparison with VirtualBox on the desktop, assuming that you choose the right front end. Aqemu offered the best performance on all tested Linux distributions and installing the virtual machine required very little input and clicking.

VMM is a viable alternative to Aqemu without any negatives. Jqemu plays to its strengths in heterogeneous environments, where admins want to launch the same interface on the virtual machine in spite of different host computers with different operating systems. But to achieve this, administrators have to plan for considerably more configuration overhead for each virtual system than with Aqemu and VMM. ■■■

INFO

[1] KVM: *http://www.linux-kvm.org*
[2] VirtualBox: *https://www.virtualbox.org*
[3] VMware Workstation: *http://www.vmware.com/us/products/workstation.html*
[4] oVirt: *http://ovirt.org*
[5] Proxmox VE: *https://pve.proxmox.com*
[6] GKVM: *http://gkvm.sourceforge.net*
[7] Qemulaunch: *https://gna.org/projects/qemulaunch/*
[8] QtEmu: *https://qtemu.org*
[9] Qemu Manager: *https://sourceforge.net/projects/qemumanager/?source=directory*
[10] Kimchi: *https://kimchi-project.github.io/kimchi/downloads/*
[11] VirtLive: *https://sourceforge.net/projects/virtlive/*
[12] Qemu: *http://wiki.qemu.org*
[13] Spice: *https://www.spice-space.org*
[14] Aqemu: *https://sourceforge.net/projects/aqemu/*
[15] Virtual Machine Manager: *https://virt-manager.org*
[16] Jqemu: *https://sourceforge.net/projects/jqemu/*



Endless OS – A PC for Emerging Countries

A PC for the People

Endless OS is a Debian-based Linux for users who might not have access to the Internet.

By Ferdinand Thommes

Even in the 21st century, some regions do not have continuous Internet access. Many of these Netless areas are in third-world countries, but even in the United States, it is not uncommon for residents in smaller towns and rural regions to lose service for hours or even days.

The creators of Endless OS believe users in areas with weak or non-existing Internet access are an underserved market. The company, which goes by the name of Endless [1], has a combined hardware/software solution designed for beginning users in environments that can't depend on the Internet.

The Endless product line consists of a Debian-based Linux distro tailored for the company's low-access use case and a collection of mini-computers to run the system. The system is designed to use an ordinary TV as a monitor.

The operating system is Linux under the hood, but, unlike most modern desktop systems, which are increasingly dependent on the cloud, Endless OS is prepackaged with a collection of tools intended for a standalone environment. According to the company, Endless OS comes with 100 preinstalled applications, videos, and lectures on academic subjects, and an encyclopedia with more than 50,000 entries.

If you are thinking Endless OS might be aimed at an educational environment, you would be correct. A video at the company's website shows the system in use with a variety of third-world classroom settings that look like Africa and South America. However, Endless OS requires way more modern conveniences than the bare-bones One Laptop per Child (OLPC) system, which came with its own foot pump for human-generated electrical power. Endless OS is intended for a user that might not have Internet access but does have fixed housing, a power supply, and a TV.

The Endless OS developers are deeply involved in the open source scene. Endless is a member of the Gnome Advisory Board [2], and the company is also active in Debian Derivatives [3]. Endless was founded in 2011 as a Kickstarter campaign [4], and it currently has subsidiaries in San Francisco and Rio de Janeiro.

Dual Solution

On the hardware end, Endless offers two mini-PCs with a spherical design. The smaller version costs $79 or $99, depending on the equipment level, and the larger version costs $189 or $229.

The smaller sphere PC, dubbed the Endless Mini, has an Intel Celeron CPU clocked at 1.5GHz with 1GB RAM, 24 or 32GB flash memory, three USB 2.0 connections, HDMI port, Ethernet, and audio. The larger Endless One comes with a 2.17-GHz Intel Celeron, 2GB RAM, a 500GB hard disk drive, Gigabit LAN, two USB 2.0 ports, and a USB 3 port. The PCs use a TV with HDMI input as a display; Endless OS is preinstalled as the operating system.

Tried and Trusted Software

The software side of the combined solution is the recently released 3.0 version of Endless OS. The developers make the

Figure 1: Endless OS's interpretation of the Gnome shell looks very different from the ones you may be familiar with in standard systems.

system available for free download independently of the alternative PC. Endless OS is based on Debian GNU/Linux 8 "jessie" and a customized Gnome desktop (Figure 1).

The 100 preinstalled applications do not require Internet access. The onboard encyclopedia is basically an offline version of Wikipedia (Figure 2). You can control the system through an App Center, which, much like the download tool for a smartphone, lets users install apps and app bundles by pressing large icons.

Endless OS is designed to be visually intuitive. Users click directly on the desktop to start an application; the menu, which most desktop environments use for orientation and navigation, is only used for accessing the settings. Thanks to a site-specific browser (SSB) [5], users can create web pages for local access from the desktop.

Network Independent

The company offers compressed Endless OS images in a basic version of about 2GB or a full version with 13GB. The unpacked image of the full version requires about 20GB space on disk, whereas the smaller version takes up just under 6GB. The full version is used on the larger of the two PCs.

The image can be transferred directly to a USB flash drive and started from there. To use it in VirtualBox, you have to convert it into a VDI image up front. The *Endless Installer for Windows* lets you download the image and requires at least Windows XP. It then creates a bootable USB stick. The new version 3.0 of Endless OS is also suitable for parallel installation with Windows. A YouTube video points out the right approach [6].

We put both versions of Endless OS through a short test. We used `dd` to copy the smaller version to an 8GB USB stick, and we converted the larger into a VDI file for VirtualBox (see the "Converting VDI" box).

Both versions launched without any problems and only differed in terms of scope: For example, the smaller variant is missing the encyclopedia, although you can install it retroactively as a Flatpak.

Well Prepared

The interface of Endless OS offers easily accessible topics such as science, sociology, health, programming (Figure 3), and many others, offering in-depth content even without an Internet connection.

LISTING 1: Convert the Base Image

```
$ VBoxManage convertfromraw eos-eos2.6-i386-i386.160602-041751.base.img
  Endless.vdi --format VDI --variant Standard
$ VBoxManage modifymedium Endless.vdi --resize 30720
```

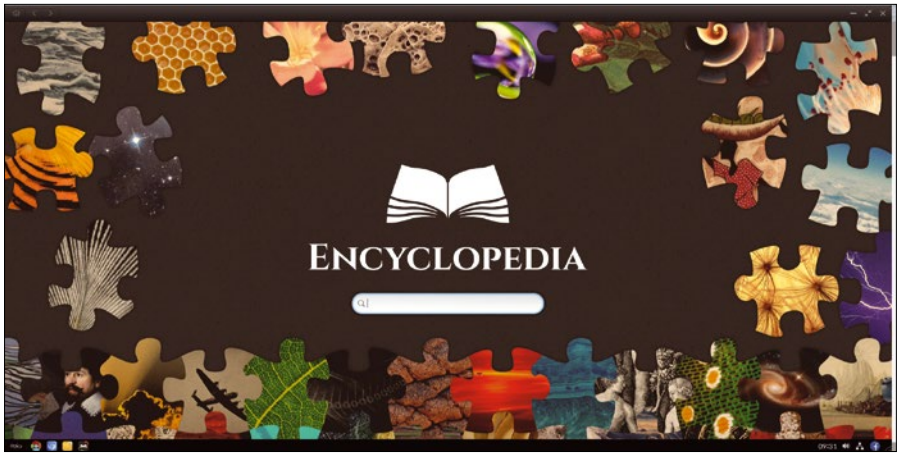


Figure 2: The encyclopedia integrated into the system also works without an Internet connection.

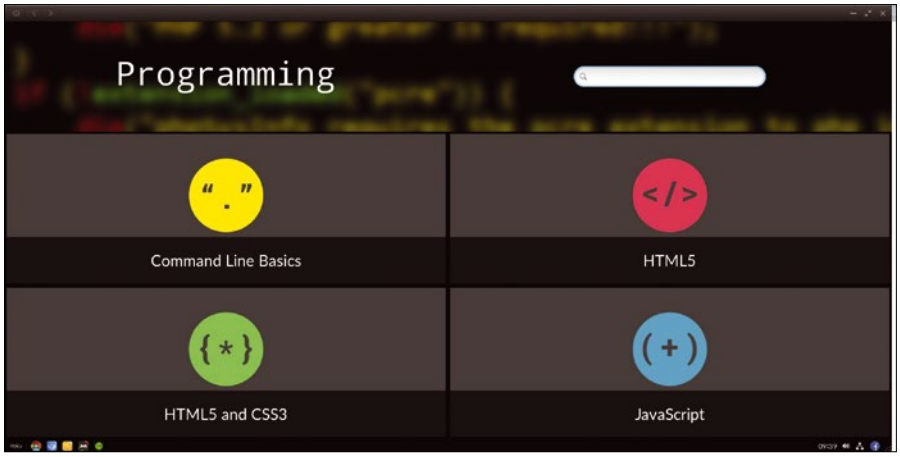


Figure 3: Offline courses for programming languages help anyone interested in learning programming, even without Internet access.

CONVERTING VDI

The following description assumes an installed VirtualBox, which Oracle provides free of charge for private use. At the command line, convert the base image using the command from the first line of Listing 1. Then, expand the space available on the virtual system with the call from the second line – in our example, we went for 30GB. Then create a virtual machine in VirtualBox, add your generated image as a virtual disk, and start the system.

There are also games and many well-known Gnome apps (Figure 4).

To make it easier to get started with the system, Endless hides many traditional desktop elements and the package management system from the user. The principle used here is very similar to that of Android or iOS, in which the operating system consists of a read-only image that can only be updated as a whole.

In the case of Endless OS, OSTree [7], which is well-known from Gnome development, is used for this purpose. Based on a guide [8], users can create an operating system from the Debian binary packages. The Endless OS system is automatically updated in the form of atomic updates that can be rolled back.

Carefully Considered

On top of the operating system, the applications are provided as packages built from Debian sources in the form of Flatpaks [9]. Endless OS is the first distribution to completely rely on this new package format. A dedicated app store provides the applications, which can be updated via the *Gnome Software* package manager. The current Gnome 3.20 from Debian "unstable" is used for this purpose, because earlier versions of *Gnome software* are unable to handle Flatpaks.

This method decouples the operating system upgrade completely from updates of individual applications. Project lead Cosimo Cecci, a Gnome developer himself, sees Endless OS as the distribution that comes closest to the Gnome OS [10] envisaged years ago but never realized. From a technical point of view, Endless OS remains very close to the original Gnome under the hood and deviates only in terms of a different shell with a unique user experience.

Conclusions

In its content, interface, and underlying technology, Endless OS has achieved something extraordinary. To our knowledge, Endless OS is the first end-user operating system to feature automatic updates that can be rolled back. In terms of administration, Endless OS is well suited for its target audience.

In many emerging countries, cheap smartphones are often the first and sometimes only contact with high-tech electronics. It thus makes sense to make the Endless OS user interface resemble a smartphone environment. A PC with 20GB or more of carefully prepared content for $100 or $200 may be a viable option for people in emerging markets looking to enter into the world of computers. ■■■

INFO

[1] Endless Computers: *https://endlessos.com*

[2] Gnome Advisory Board: *https://wiki.gnome.org/AdvisoryBoard*

[3] Debian Derivatives: *https://wiki.debian.org/Derivatives/CensusFull*

[4] Endless OS Kickstarter: *https://www.kickstarter.com/projects/1381437927/endless-computers/description*

[5] SSB: *https://en.wikipedia.org/wiki/Site-specific_browser*

[6] Dual boot video tutorial: *https://www.youtube.com/watch?v=sl-7PWKwmfE*

[7] OSTree: *https://ostree.readthedocs.io/en/latest/*

[8] OSTree builder: *https://github.com/dbnicholson/deb-ostree-builder*

[9] Flatpak: *http://flatpak.org*

[10] Gnome OS: *https://wiki.gnome.org/GnomeOS*

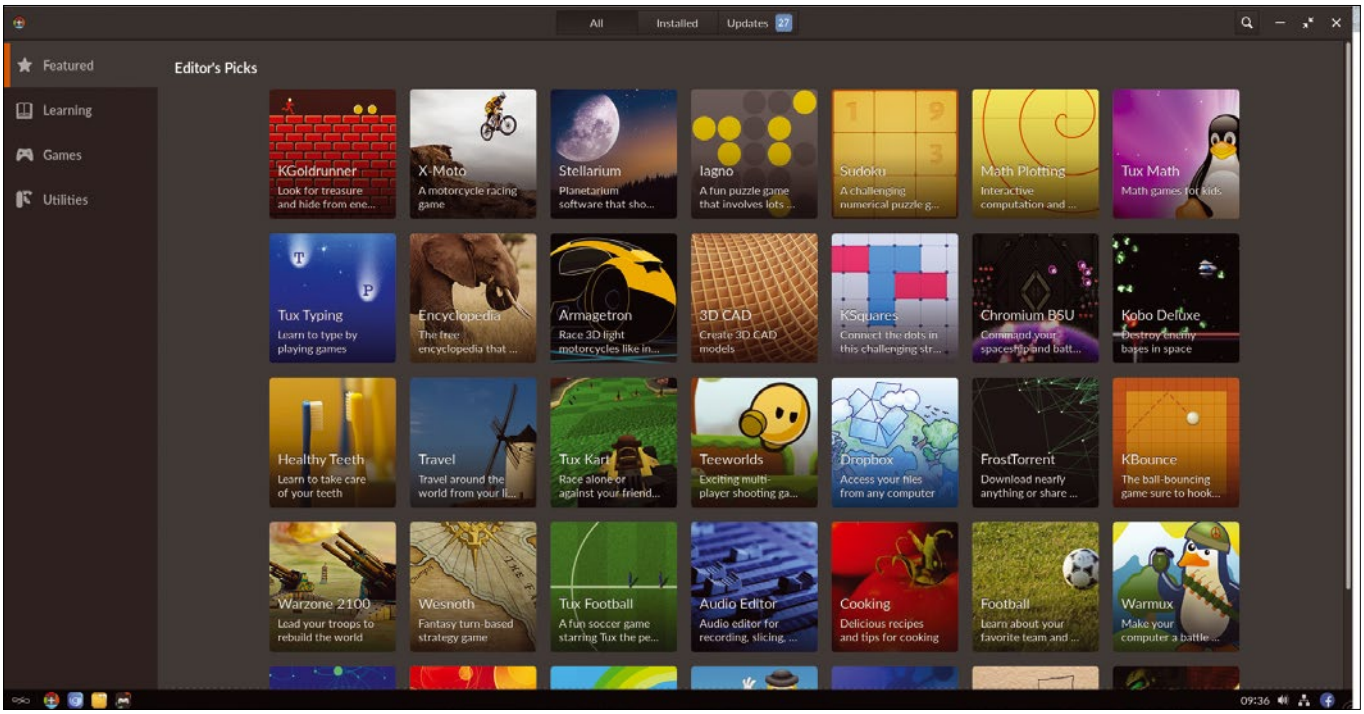
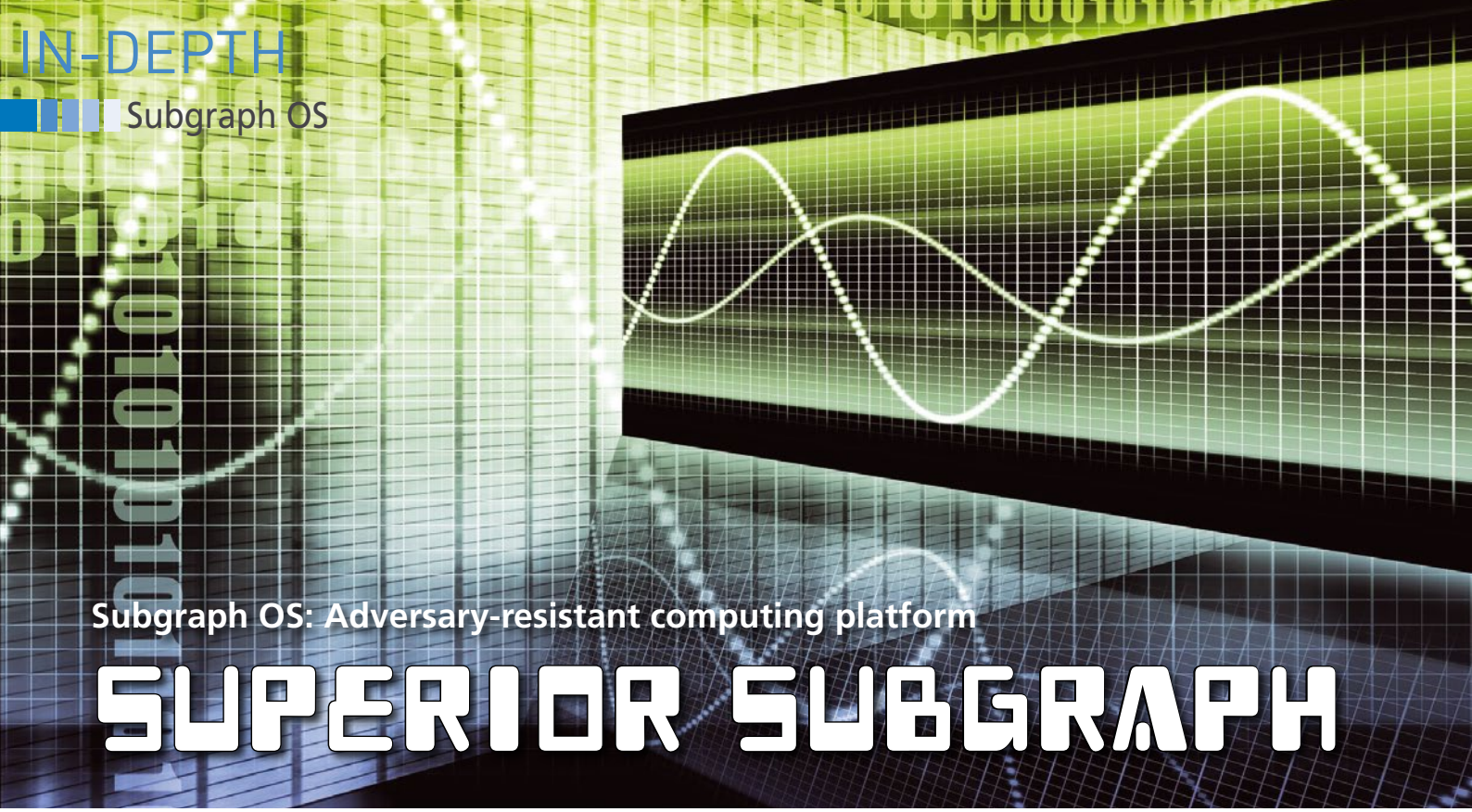


Figure 4: Endless OS comes preinstalled with a variety of programs for various purposes. *Gnome Software* is the application manager.

Subscribe now!

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

shop.linuxnewmedia.com/subs



Subgraph OS: Adversary-resistant computing platform

SUPERIOR SUBGRAPH

Kid-tested and Snowden approved – is Subgraph, the privacy-oriented OS, now ready for humans? *By Nate Drake*

In early 2016, David Mirza Ahmad, president of Subgraph, announced their OS as a public alpha. The announcement took place at the Logan CIJ Symposium, which is dedicated to fighting surveillance and censorship, and was greeted warmly by Edward Snowden himself [1].

Ahmad also advised, "The Internet is more hostile than it's ever been. Subgraph is addressing that problem."

Since then, he and the rest of the four-man team in Montreal have been devoting themselves to developing Subgraph. Most recently, their efforts have culminated in Subgraph Alpha r3 with a range of news apps and security features [2] (Figure 1). The project is backed financially by the US government-sponsored Open Technology Fund, which is also behind privacy-oriented distros like Qubes OS and Tails.

The similarity doesn't end there. Subgraph protects users through a hardened kernel, a carefully selected list of apps, and anonymizing network connections.

Starting Subgraph

Subgraph r3 can be downloaded from the project's website [3]. In keeping with the strong emphasis on security, the 1.3GB ISO download is accompanied by a SHA sum and GPG signature, which you can use to check the integrity of the image before copying to DVD or USB.

The team also cautions that this is still alpha software, so it should not be relied upon for any serious project.

That said, it's clear that unlike many privacy-oriented distros, the Subgraph team has emphasized usability as well as privacy. The GUI is the familiar Gnome Desktop Environment running on a modified version of Debian 9 (Stretch). This means that the installer will pose no issues if you've ever installed a Debian-based system. The major difference is that encryption of your drive via Linux Unified Key Setup (LUKS) is mandatory.

Installs require at least 20GB of disk space and a minimum of 2GB of RAM, although 4GB is recommended. If you prefer to run Subgraph in Live mode, at

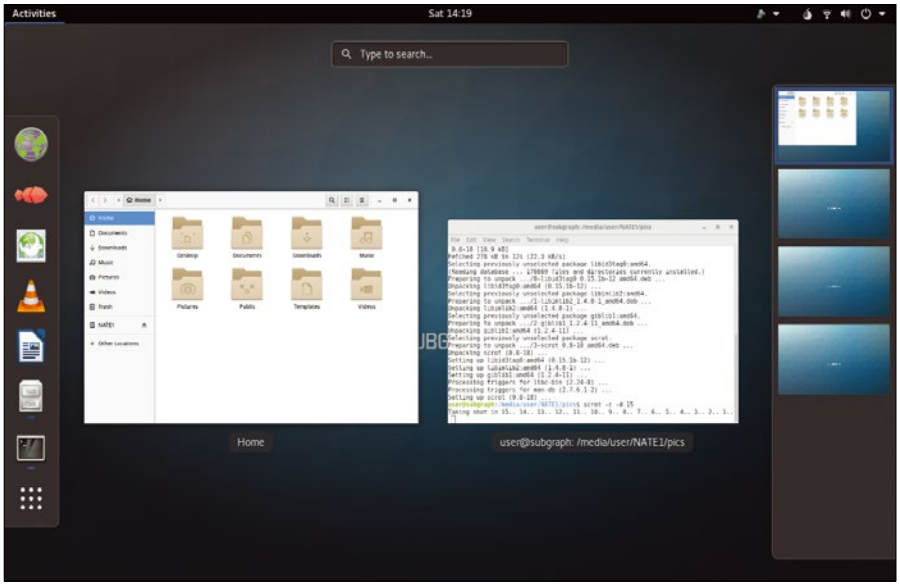


Figure 1: Subgraph's overall look and feel is very similar to Debian 9, but there are new apps and hidden features.

AUTHOR

Nate Drake is a freelance journalist specializing in cybersecurity and retro tech.

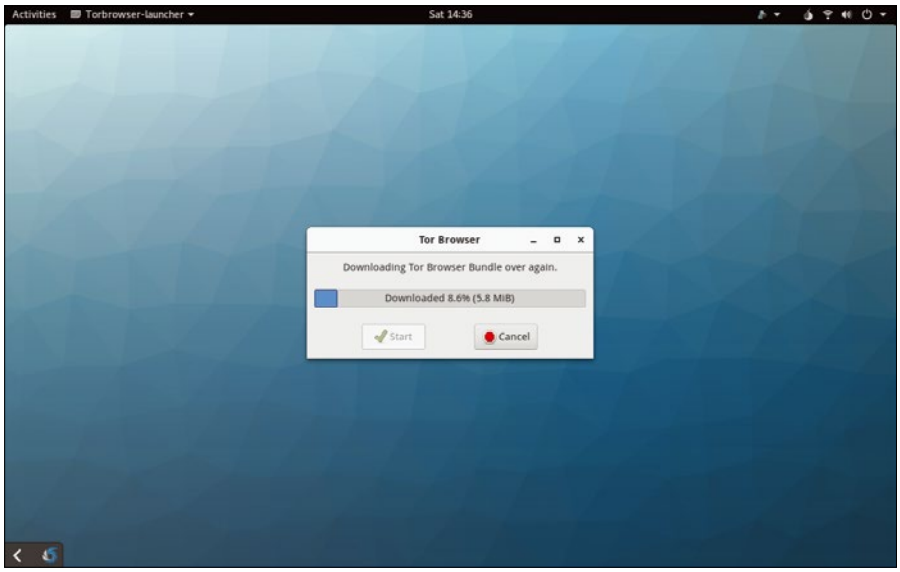


Figure 2: Subgraph checks the signature for the download and starts again, if necessary, to make sure the Tor Browser isn't compromised.

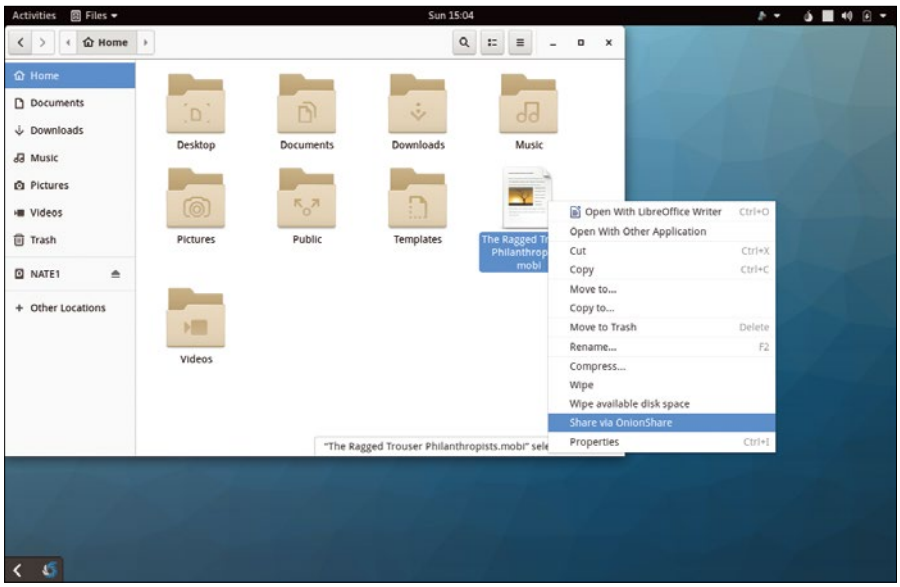


Figure 3: Right-click on any file to share via OnionShare. The app will launch automatically with the link to the file.

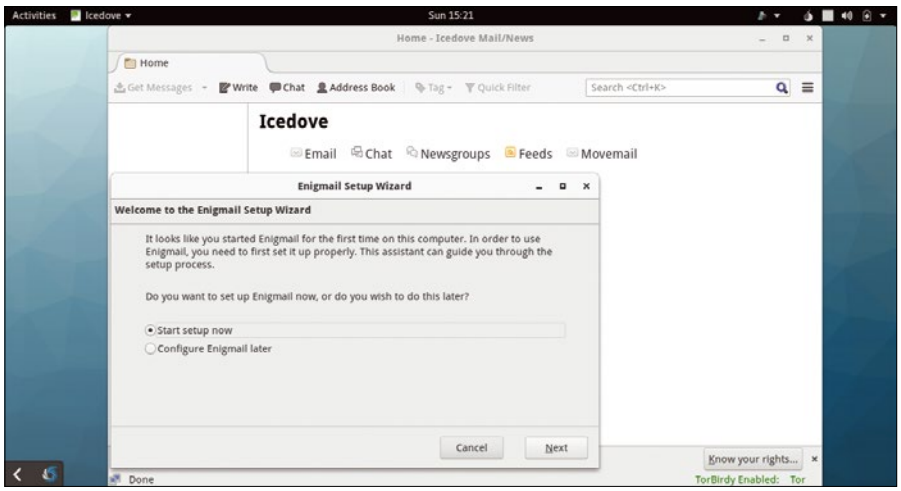


Figure 4: Subgraph includes the Icedove email client. The plugins Torbirdy and Enigmail are preinstalled to anonymize your connection and encrypt your emails, respectively.

least 4GB of memory is required. Although these requirements are onerous, the minimum amount of RAM required for installation is the same as for Qubes OS and Tails.

Currently only 64-bit machines are supported. On first boot, the OS superficially seems to resemble a stock install of Debian, albeit with a few new preinstalled apps. Under the surface, however, Subgraph has some marked differences.

Torifying Apps

Opening the Gnome Shell Dash reveals the stock Subgraph apps. A good starting point is the system's default Tor Browser, which helps to anonymize your connection while browsing, as well as hugely reduces the chance of browser fingerprinting.

The browser also contains a slider bar, which allows you to change the level of security used at the expense of loading certain types of web pages.

On first run, Subgraph downloads a tarball of the browser and uses signature verification to make sure that the integrity of the file has not been compromised (Figure 2). During testing, the installation failed; however, the developers' GitHub page revealed that they were aware of this and that there's a workaround [4].

Aside from the Tor Browser, the preinstalled app OnionShare can also connect directly to Tor hidden services. It is specifically designed for file sharing. The advantage of using hidden services via a `.onion` address is that both the sender and receiver are hidden. Because the traffic never leaves the Tor network, there's no way to monitor entry and exit points for vulnerabilities, so obtaining metadata about files you share is virtually impossible. You can share files via OnionShare with ease from within the Nautilus File Manager, simply by right-clicking on them and choosing *Share via OnionShare* (Figure 3). The OS incorporates IceDove, which is an unbranded version of the Mozilla Thunderbird email client. Incoming and outgoing mail is routed through the Tor network thanks to the pre-installed *TorBirdy* plugin. IceDove also comes with the Enigmail plugin to allow you to send and receive gpg encrypted emails (Figure 4).

Subgraph also comes with the "torified" instant messenger Ricochet. This privacy-minded app from the invisible.

im team uses Tor hidden services to allow chat users to connect directly to one another, avoiding the risk posed by a faulty or malicious central server.

For security reasons, all of these apps run inside their own sandboxes (more on this later).

Marvelous Metaproxies

As handy as privacy minded apps can be, not all useful Linux applications are specifically designed to be used over Tor. Sufficiently skilled users can sometimes manually configure applications capable of connecting via proxy to use the Tor network, but this can be tricky to set up correctly. Any application leaking data while you're using Tor can potentially be used to trace your location and access your data.

Subgraph OS resolves this issue by routing all outgoing connections that otherwise wouldn't go through Tor via a Subgraph Metaproxy. This ensures all connections are made via the Tor network. However, crucially, programs, such as the Tor Browser Bundle, that already use Tor are ignored by the Metaproxy.

Another extremely well thought-out Subgraph feature is the inclusion of the control port filter `ROFLCopTor`.

By default, the Tor service is managed by a control protocol, which regulates information about Tor connection, starts hidden services, and changes your configuration. Most programs don't need access to all these settings.

`ROFLCopTor` acts as a proxy server between Tor control clients the Tor control server port. It has a number of built-in policies in place to filter incoming and outgoing commands on an application-by-application basis to determine which features they can access. This substantially reduces the chance that a compromised program could de-anonymize your connection or otherwise be used to spy on you.

Your privacy is increased even further by `Macouflage`, which creates random network addresses for all your interfaces, giving you better anonymity even when connecting to the same networks.

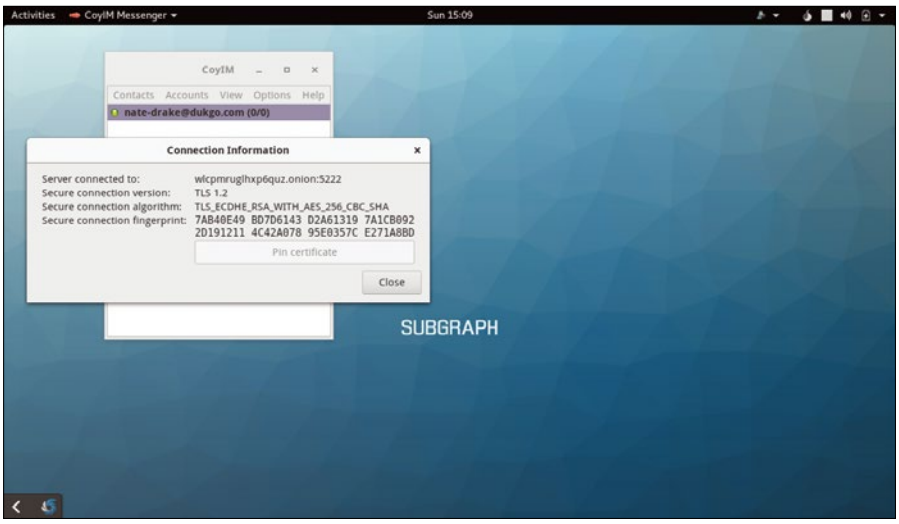


Figure 5: Click *Register* to create a new account with an XMPP server that supports Tor hidden services, such as DuckDuckGo or Calyx.

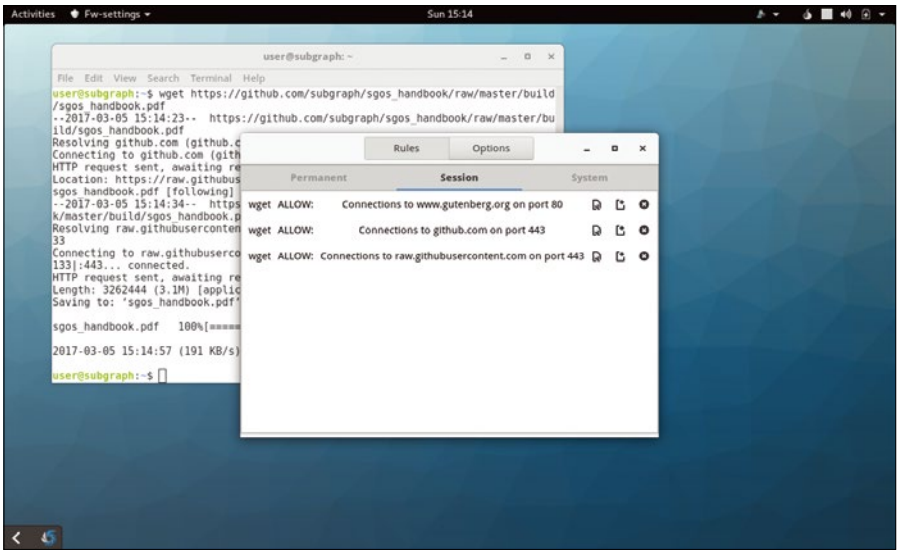


Figure 6: The Firewall allows you to fine -tune settings for individual programs. Here the command-line utility `wget` is allowed to connect on both ports 80 and 443 for this session.

Playing Coy

Special mention must go to Subgraph's own CoyIM instant messenger. CoyIM is designed to work with the *XMPP* message protocol, so you'll need to register an account with a compatible chat server to use it. (Figure 5). Connections are made via Tor + TLS to clearnet chat servers, although for extra security you may wish to consider using an XMPP provider that supports connecting via Tor hidden services, such as The Calyx Institute [5]. Where possible, the messenger will use the hidden service for a chat server, if it has a record of it.

All conversations are encrypted via Off the Record (OTR) Messaging, which offers perfect forward secrecy – in other words, if the keys are compromised and a single conversation is decrypted, this won't help decrypt any of your other chats. On first run, CoyIM will ask you if you wish to encrypt your configuration file and if so, to choose a master password. You'll then be asked to either sign in or register your XMPP account. A green dot will appear next to your name once you're connected.

Like Subgraph itself, CoyIM has yet to undergo a full security audit, so use with caution.

Sub Rosa

Aside from hard work and a pat on the proverbial back from Snowden, Subgraph has much to offer the privacy-minded user.

Subgraph's website describes the OS as an "adversary resistant computer platform." The term adversary seems to have been interpreted broadly to include both hackers and government officials from oppressive regimes.

Users will notice this approach most when new programs are run: Subgraph

Firewall will ask you to *allow* or *deny* the connection (Figure 6). You can also choose to permit a connection for only a certain amount of time. The default firewall policy of most Linux distros is to block incoming connections but allow all outgoing ones, so it's heartening to see that Subgraph polices connections on an application-by-application basis so seriously.

The Subgraph team discovered an excellent case in point when coding their firewall with Gnome Calculator. By default, this program will connect to various financial institutions online to gather currency exchange rates. This seems fairly harmless until you consider that searching for a specific currency could be used to determine which country you're in or about to travel to.

The Linux kernel used by Subgraph's latest release has been hardened with GrSecurity v4.8.15, which contains a number of privacy-related enhancements such as USB Lockout, a background process that denies all access to USB devices when the screen is locked or you're logged out.

GrSecurity also includes PaX – a set of patches that makes the kernel and user-space much less vulnerable to memory corruption exploits.

One form of memory corruption exploit, known as "stack smashing," is among the oldest and most reliable ways to hack into a system, whereby an adversary can execute arbitrary malicious code on your system.

PaX protects your computer from memory corruption exploits in a variety of ways. Chief among these is randomizing the layout of process memory, which makes it much more difficult for hackers to locate exactly where they've placed malicious code. It can also make memory pages non-executable. Payload stored in non-executable memory cannot run.

PaX also attempts to proactively kill applications that violate its security policies, hopefully preventing issues before they even arise [6].

The Wonderful Wizardry of Oz

Subgraph OS runs desktop applications inside the team's very own security sandbox named Oz. This builds on the protection offered by kernel hardening to protect your system from compromised applications.

On most mainstream Linux distributions, desktop applications running with the X11 display server can see and interact with each other. The Oz sandboxes prevent this by using Xpra, which renders individual applications with their own display server, so they cannot interact, improving security.

Oz can also restrict access to certain files, as well as network access and audio playback for applications that don't need it.

The actual sandboxing process works seamlessly for the user. Applications that are to be run in Oz are renamed, and a symbolic link is created in the original location of the program, which directs the system instead to the Oz binary. When the program is started, Oz automatically scans the program name and examines its associated policy document, which governs how the application launches and its runtime environment.

The Oz daemon then creates the sandbox and launches the real application

safely inside it, as per the policy document. Outside the sandbox, an Xpra instance connects to the Xpra server inside the sandbox. This means the application cannot log keystrokes or otherwise meddle with applications [7].

The applications that run within the Oz sandboxes are those most likely to be exposed to untrustworthy networks and data, such as the Tor Browser and Evince PDF Reader. You can, however, enable Oz profiles for other applications by running the command:

```
sudo oz-setup install <program-name>
```

Black and White

Oz also further bolsters your protection from exposed system calls through using `seccomp`. For non-Linux die-hards, system calls are simply the way that an application requests a service from the OS kernel.

These can be exploited by malicious people, whereby a program is run by a normal, unprivileged user, who then spawns a root shell, giving the user full access to the system. The Semtex exploit of 2013 is an excellent example of this.

Oz assigns policies on a per-application basis. Programs are killed if they violate the policy.

If a system call within an attacker's payload (such as one that would unnecessarily require root-level privileges) is not explicitly permitted, it will not be allowed.

Oz supports both blacklist and whitelist policies. For all of Subgraph's own supported apps, Oz maintains a white list, which blocks all system calls except those which are explicitly allowed.

Oz has a generic blacklist that automatically blocks more dangerous or unusual system calls, located in `/var/lib/oz/cells.d/generic-blacklist.seccomp`.

The latest version of Subgraph includes a new *Go seccomp-bpf* library developed by the ThoughtWorks Tiger team to help create much more efficient policies for sandboxed apps. The Subgraph handbook's appendix maintains a complete list of system calls in the OS, which you can use to create your own policies if you wish.

Subpar

Subgraph's sandboxing features can cause problems for certain applications. PDFs, for instance, which are notoriously vulnerable to exploits, are opened by the built-in Evince document reader, which cannot access the Internet or any other file besides the PDF it is currently reading.

If Evince is already open, you can click on Oz's icon, which resembles a stylized zebra, at the top right of the screen and then *Add File* to open one or more files. Subgraph allows you to make them read-only. You can use the Oz menu to add files inside the Tor Browser in the same way (Figure 7). This is useful if you actually want to upload files. You do not need to do this for OnionShare as the process is handled automatically when you choose to share a file from within Nautilus.

Oz also makes a special exception for the Tor Browser sandbox when it comes to downloads to prevent all files being lost each time you close the browser. This is done through a shared directory located in `~/Downloads/TorBrowser`. Any files saved there while surfing will be accessible after you close down the browser.

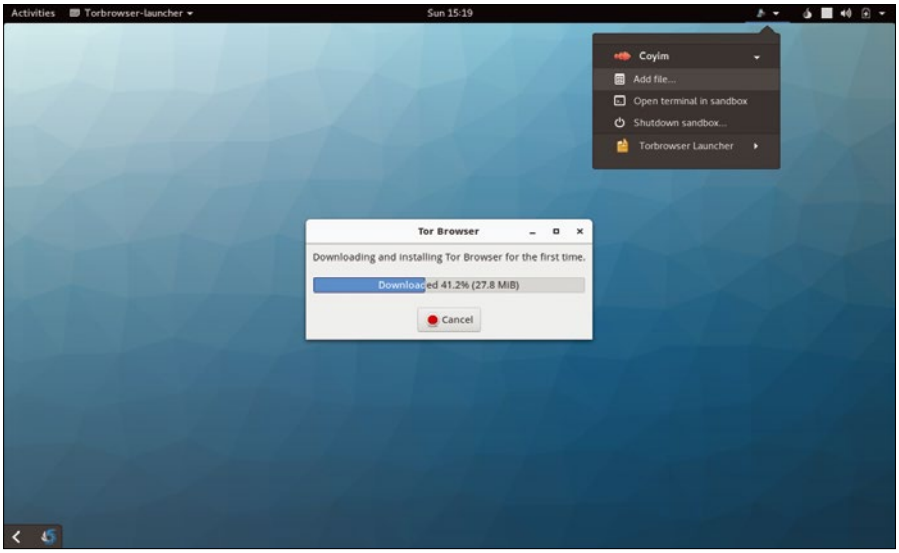


Figure 7: Click on the Oz menu to add files to applications for viewing and/or upload.

Final Subgraph

Overall, Subgraph is an excellent proof of concept. While the developers stress it's not yet ready for privacy applications, it certainly deserves a ringing endorsement for its careful selection of privacy-related apps, Oz's ingenious sandboxing, and the crafty CoyIM.

David Ahmad was kind enough to speak to Linux Magazine about some of the upcoming features in Subgraph. Chief amongst these is an Oz sandboxed version of the Chromium Browser which can visit regular 'clearnet' websites.

Any readers who are impatient to give this new setting a try can follow the steps on the Subgraph website [8] to install the browser manually. Watch this space for further developments.

While you should rely on the above two privacy distros for now, watch this space for further developments from the Subgraph project. ■■■

INFO

[1] Subgraph will be Snowden's OS of choice, but it's not quite ready for humans yet: *https://thenextweb.com/insider/2016/03/16/subgraph-os-will-snowdens-os-choice-not-quite-ready-humans/#gref*

[2] Subgraph OS December 2016 ISO availability: *https://subgraph.com/blog/subgraph-dec2016-iso-availability/*

[3] Download Subgraph OS Alpha r3: *https://subgraph.com/sgos/download/index.en.html*

[4] GitHub – signature verification failed: *https://github.com/subgraph/subgraph-os-issues/issues/205*

[5] The Calyx Institute: *https://www.calyxinstitute.org/projects/public_jabber_xmpp_server*

[6] The PaX Team homepage: *https://pax.grsecurity.net/*

[7] Oz technical details: *https://github.com/subgraph/oz/wiki/Oz-Technical-Details*

[8] Clearnet Sandboxed Chromium: *https://subgraph.com/sgos/documentation/clearnet-chromium/*

Lead Image © Alphaspirit, 123RF.com

Converting font formats

Quick Change

Perform trouble-free font conversions from the command line with an easy-to-use script.

By Bruce Byfield

Ordinarily, few users have to concern themselves with converting file formats for fonts. However, with LibreOffice 5.3 dropping support for Type 1 fonts [1], conversion becomes relevant, especially if other Linux applications do the same. Users are suddenly faced with either finding a solution or losing altogether fonts collected over years, some of which may be irreplaceable if they are from small font designers. Moreover, if support for Type 1 fonts is dropped, in a few years TrueType fonts may also be dropped in favor of OpenType fonts. Fortunately, a number of solutions are available, including a batch script in FontForge [2], a free software font creation tool included in most distributions.

Type 1 fonts (`.pfb`), also known as PostScript fonts, were first released in 1985. As might be expected with such an old format, they include some major limitations. Information in Type 1 fonts is contained in several files (`.pfb`, `.afm`, and, on Windows, `.pfm`), and older Type 1 fonts are short of accents and other diacritical marks. Moreover, glyphs (characters) are divided into one-thousandths – less than half of the highest-quality TrueType fonts – and are less efficient than TrueType at hinting or rasterizing. All the same, Type 1 fonts are superior to even older font formats and remain good enough for professional uses.

In theory, TrueType fonts, which were introduced a few years after Type 1 fonts, allow more detail. However, in the early 1990s, hastily created TrueType fonts gave the format a bad reputation, causing many designers at the time to avoid it. As a result, small font foundries often chose to use Type 1, which means that many rare fonts are unavailable in TrueType, a fact that has helped to keep the format alive.

In 1996, Microsoft and Adobe released the OpenType format (`.otf`) and continued developing it over the next 15 years. Built on TrueType, OpenType is an advance on both Type 1 and TrueType, in

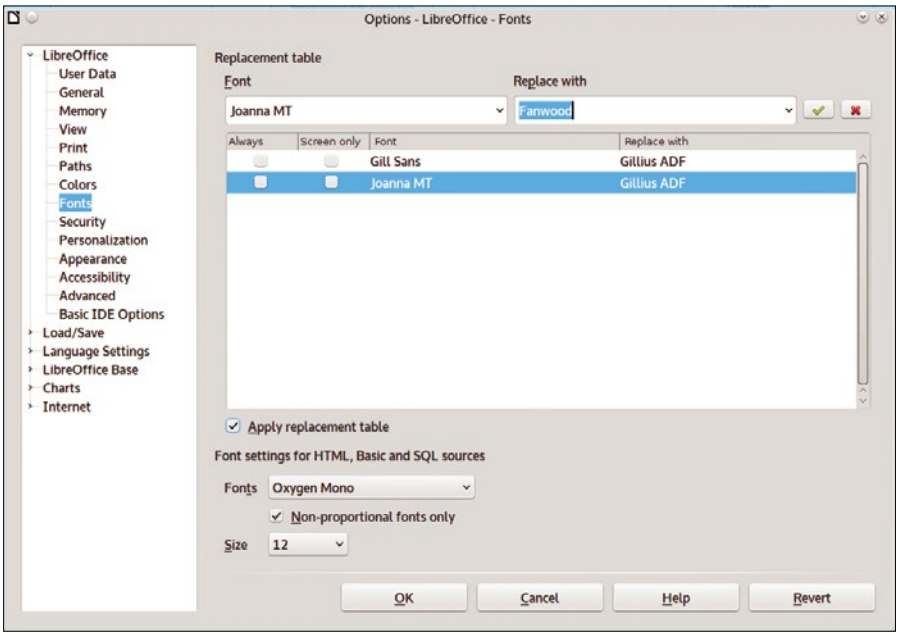


Figure 1: Setting up replacements for fonts is one alternative to conversions but may still require reformatting.

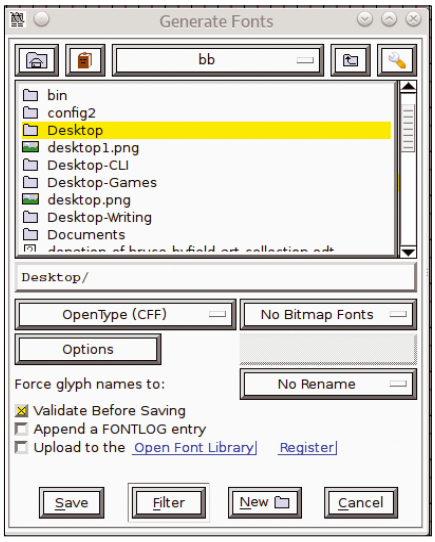


Figure 2: Single font conversions can be done from the FontForge interface.

that it includes support not only for a full range of diacriticals but also for other advanced typographical features, such as ligatures, small capitals, or old style figures in a single file. For the purposes of conversion, OpenType's most important feature is that its vector font outlines can be drawn from either Type 1 or TrueType fonts. With these and other advantages, OpenType has gained steadily in popularity ever since its release, and it has been the font format of choice for well over a decade.

The change in LibreOffice 5.3 is due to the decision to use HarfBuzz [3], a new and advanced font rendering engine that does not include Type 1 support. Under Windows, this decision has little effect, because Microsoft Office 2013 and many other Microsoft products, apparently including at least some recent versions of Windows, have already dropped Type 1 support. However, in Linux, which continues to support the Type 1 format, the effect is more far-reaching. Not only is the dropping of Type 1 support an inadequately announced step backward, but it also places LibreOffice out of sync with other applications, including GIMP, Krita, Inkscape, and Scribus. Consequently, transferring a file from one application to another has become more difficult.

Figure 3: Another way to convert only a few fonts is from FontForge's online tool.

Limited Solutions

Because HarfBuzz shows no signs of adding Type 1 support, users who wish to continue using Type 1 fonts must look elsewhere for solutions. At average prices of $10-75 per typeface for OpenType font versions, replacing a library of Type 1 fonts may be too expensive to be an alternative.

One solution is to set replacements for Type 1 fonts under *Tools* | *Options* | *LibreOffice* (Figure 1). Using free-licensed fonts is especially useful, because they are likely already to be in `.otf` format. In some cases, the free fonts may be metrical equivalents of Type 1 fonts; characters in the Liberation fonts, for example, occupy the same space as Times Roman, Arial, and Courier, so that no reformatting is necessary. However, many replacements will not be so exact.

Other solutions include switching to Apache OpenOffice, to remain with pre-5.3 version of LibreOffice, or to use Calligra Suite's Writer. However, all these mean abandoning past – and future – improvements in LibreOffice. Moreover, any of these solutions seem temporary at best. Adopting one could mean having to face the same problem again in a few years as backward compatibility ceases, small projects disappear, or HarfBuzz becomes more prevalent.

All things considered, converting fonts to OpenType format seems the most practical solution --especially if the tedium of conversion can be alleviated by running batch scripts. The Adobe Font Development Kit for

LISTING 1: Sample Script

```
01 #!/usr/local/bin/fontforge
02 i=1
03 format=".otf"
04 while ( i<$argc )
05   if ( $argv[i]=="-format" || $argv[i]=="--format" )
06     i=i+1
07     format = $argv[i]
08   else
09     Open($argv[i])
10     Generate($argv[i]:r + format)
11   endif
12   i = i+1
```

TABLE 1: Explanation of Script

Excerpt	Description
`#!/usr/bin/fontforge`	This comment calls on FontForge to execute the script. The location of the FontForge binary may be in a different place in your distribution, such as `/usr/share/bin/fontforge`.
`i=1`	Creates a local variable called `i` with a value of 1.
`format=".otf"`	Sets a default conversion format of OpenType. This format can be changed by setting the format option when running the command (see below).
`$argc`	Variable for the number of arguments passed to the script.
`$argv[i]`	The array containing the argument passed to the script.
`( $argv[i]=="-format" || $argv[i]=="--format" )`	Creates a format option. Notice that the both `-format` and `--format` (one or two hyphens) are valid options.
`if ..else`	If `-format` or `--format` is specified, then that format is used. If neither `-format` or `--format` is used, then the default set in `format=` is used.

OpenType includes the tx script for Type 1 to OpenType conversion [4], but installation instructions for Linux are obscure and require a knowledge of how to add paths to an environment, making them difficult for many to use. Similarly, although FontForge supports Python scripts as well as its own scripting language, so far no one seems to have written one for conversion.

FontForge Solutions

FontForge is a powerful but slow application with a primitive-looking interface that, in the best free software tradition, is the result of one developer's enthusiasms – in this case, the enthusiasm of George Williams, who developed FontForge as a retirement hobby before passing its development on to other hands in 2012. Although it's less sophisticated-looking than proprietary font design applications, FontForge remains a full-featured tool, and, according to Wikipedia, it was used to develop such free-licensed fonts as Cantarell, DejaVu, Inconsolata, and Linux Libertine. Most major Linux distributions include FontForge in their repositories [5].

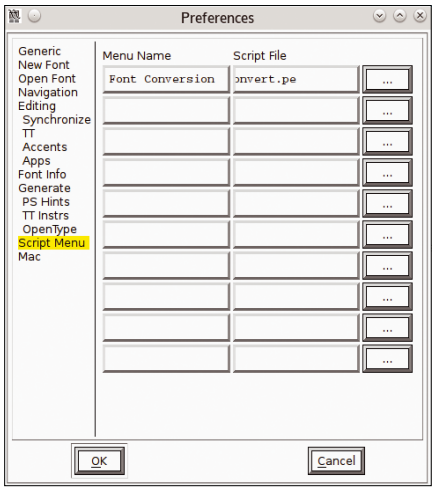


Figure 4: You can run the conversion script from a command line or from the FontForge interface.

FontForge offers two solutions for converting single fonts. First, from the FontForge interface, you can open a `.pfb` or `.ttf` file, and then select *Font* | *Generate Font* (Figure 2), and set the export format to OpenType (CFF). Second, FontForge maintains an Online Font Converter [6], in which you select the export format and then drag and drop files to the window (Figure 3).

These solutions are practical if you have only a few fonts to convert or else are content to convert files only as needed. However, if you have dozens or hundreds of Type 1 files and want to get the entire process over and done with, the most practical solution is to run a headless batch script – one that calls upon FontForge without opening the graphical interface. Happily, the FontForge website includes an example of such a script that describes its construction, adding complexity step by step [7]. With one or two minor modifications, the script samples remain usable today (Listing 1). Table 1 parses the file to show what each element does.

To run a batch file, run

```
chmod +x convert.pe
```

to give it the necessary permissions to run. Place all the files you want to convert in the same directory, and then switch to the directory and run:

```
convert.pe --format "EXPORT-FORMAT"➥
   *.ORIGINAL-FORMAT-FILES
```

For instance, to convert Type 1 fonts to OpenFace, you would run:

```
convert.pe --format ".otf" *.pfb.
```

FontForge does not have a verbose mode, so the only evidence of success is the appearance of the converted files in the directory.

The FontForge site also gives a version of this script for converting TrueType fonts to Type 1, but, these days, few people are likely to want that variant. Similarly, the same page describes how to add more accented characters to a Type 1 font, but if you have survived without the extra accents in the past, you probably can continue to do without them. Besides, the newest Type 1 fonts already have a reasonable array of accented characters for Western European languages.

After Conversion

This script runs without problems from a command line (Figure 4). However, if you are working from the FontForge interface, you can click *File* | *Preferences* | *Scripts* to add a script menu and then add the script as a menu item.

No matter how you run the script, the conversion should be trouble-free. You will have to add the converted versions of the font to your system or to LibreOffice to use them, and, if LibreOffice is open, close, and reopen it to gain access to the files. ■■■

INFO

[1] Type 1 fonts in LibreOffice: *https://bugs.documentfoundation.org/show_bug.cgi?id=104701*

[2] FontForge: *https://fontforge.github.io/en-US/*

[3] HarfBuzz: *https://freedesktop.org/wiki/Software/HarfBuzz/*

[4] Type 1 to OpenType conversion script: *https://www.adobe.com/devnet/opentype/afdko/topic_overview.html#a_programs*

[5] FontForge distros: *https://en.wikipedia.org/wiki/FontForge*

[6] Online Font Converter: *https://onlinefontconverter.com/*

[7] Script tutorial: *https://fontforge.github.io/scripting-tutorial.html*

Hybrid image mode for Knoppix 8.0

Bye Bye DVD?

As DVD drives disappear from modern notebook computers, Knoppix offers a hybrid image mode that works for both DVDs and flash drives. *By Klaus Knopper*

The DVD is a comparatively long-lasting (50+ years) optical storage media. Yet in today's notebooks, you will rarely find a built-in DVD drive. This may be related to the unpleasant form factor, considering the DVD's capacity of about 4GB single-sided or 8GB double-sided, which offers much less storage capacity than a modern tiny microSD card. For booting from removable media,

TABLE 1: DVDs vs. Flash Drives

Issue	DVD Image File	(Flash) Disk Image File
Size (bytes, capacity)	Image size matches contained data	Image size reflects flash medium/partition size(s)
Transfer to medium	Easy to just burn to medium using standard DVD writing tools (`cdrecord`, `growisofs`, `wodim`)	Easy to write for Linux users (`dd`, `cp` to device file directly), but Windows users needs special software for accessing drives in raw mode (Disk Imager). See the "Flashing Disk Images" box
Size Limits	Approximately 4.5GB single layer, 9GB double layer	No limit (but may need higher sector size or GPT partition table for disks >= 3TB)
Filesystem	ISO 9660 read-only, Rock Ridge/Joliet extensions support Unix attributes.	Various filesystems, such as FAT (max. 4GB contained file size limit), NTFS, ext, ReiserFS, etc., may contain several filesystems (one per partition)
Partitioning	(Usually) no partition table, uses full disk, variable size up to full capacity	HD partitions (MS-DOS partition table with four primary and unlimited logical partitions, or GPT partition table, or both), fixed size of partitions in partition table, counted in sectors
Access to data without physical medium	Can be mounted via loopback device (`mount -o loop imagefile.iso mountpoint`) on Linux, or Windows Image Explorer tools	Mount via loopback with calculated "offset" option for each partition on Linux, or detect partitions with `kpartx -a file.img` and mount them separately. Windows?
Boot Code	El Torito bootable CD specification, Volume Boot Descriptor embedded in filesystem at offset 32k	First stage bootloader in MBR and/or partition boot record

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.

```
00000000   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00000010   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00000020   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00000030   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00000040   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00000050   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00000060   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00000070   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00000080   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00000090   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000000A0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000000B0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000000C0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000000D0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000000E0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000000F0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00000100   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00000110   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00000120   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00000130   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00000140   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00000150   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00000160   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
---  KNOPPIX_V7.7.1DVD-2016-10-22-EN.iso       --0x0/0x115CFE800-------------
```

Figure 1: First 32KB in ISO 9660 filesystem are set to zero.

FLASHING DISK IMAGES

Linux: `cp knoppix.iso /dev/sdb`

Windows: `Win32DiskImager.exe <click, click, click> ...`

the easily portable USB flash pen with a practical keyring connector has long been a favorite for most users instead of the CD or DVD. So, wouldn't it be appropriate for Knoppix to just give up on the old-fashioned DVD and start distributing only flash disk images instead?

As much as this seems to be a reasonable step, there are still pros and cons to both formats (Table 1).

Probably the most important issue is the easy handling of distributed images. A DVD ISO, containing the ISO 9660/Rock Ridge/Joliet filesystem, is always exactly as big as the included (unpartitioned) data; there is no adjustment needed when burning.

A hard disk or flash disk image, however, contains a partition table with fixed values for sector starts and ends of partitions. An image with a partition table for a 7.8GB drive will cause data corruption sooner or later when installed on a 7.6GB drive, since the filesystem of the last partition extends beyond the physically available space. At least, the last partition would have to be expanded or shrunk to match the physical end of data. Installing a 8GB image on a 1TB hard disk would leave the remaining unpartitioned space unused. It's quite inflexible.

The `flash-knoppix` tool circumvents this limitation when creating a bootable flash disk by first partitioning the target disk for using all available space and then creating filesystems that match the partitioning before copying over all files.

Volume Records vs. Disk Partition Tables

By its standard, the ISO 9660 filesystem used for DVDs does not use the first 32KB for anything; these bytes are usually set to zero (Figure 1), after which a volume descriptor is located at offset `0x8800` that starts with `CD001`.

Figure 2 shows the El Torito specification [1] located after the first 32KB set to zero and indicates the boot system ID.

Several of these volume descriptors can be found in the ISO 9660 image, but this one is also a boot record, which contains the boot catalog's address encoded in little-endian format at byte 71 to 74 from the records offset. This referenced boot catalog then points to the isolinux boot loader, which loads the kernel and initial ramdisk.

As you can see, the procedure of the computer's BIOS booting the ISO 9660/El Torito way is a little more complex than booting from a Master Boot Record (MBR) disk.

Now just doing a 1:1 copy of the ISO image to flash disk will not create a bootable medium, since the BIOS does not even attempt to search for ISO volume descriptors or boot records for media other than CD or DVD. Additionally, the standard sector size of a CD/DVD is specified as 2048 bytes, whereas for disks the sector size is 512 bytes or 4096 bytes, so all offsets given in blocks or sectors would be wrong.

When booting with the MBR method from flash or hard disk (Figure 3), the BIOS will read the first 446 bytes of the disk (starting at sector `0`). These first bytes contain a very tiny boot code that usually just reads the partition table at byte 446 and redirects to the next boot record at the start of a system partition marked as `bootable`, or to an offset that contains a secondary stage boot loader.

Right after the boot code, from byte 446 to 511, the MS-DOS style partition table follows, which defines the four primary partition offsets, types and

```
000086E0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000086F0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00008700   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00008710   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00008720   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00008730   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00008740   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00008750   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00008760   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00008770   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00008780   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00008790   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000087A0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000087B0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000087C0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000087D0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000087E0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000087F0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00008800   00 43 44 30  30 31 01 45  4C 20 54 4F  52 49 54 4F  .CD001.EL TORITO
00008810   20 53 50 45  43 49 46 49  43 41 54 49  4F 4E 00 00   SPECIFICATION..
00008820   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00008830   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00008840   00 00 00 00  00 00 00 FE  02 00 00 00  00 00 00 00  ................
---  KNOPPIX_V7.7.1DVD-2016-10-22-EN.iso       --0x86E0/0x115CFE800-----------
```

Figure 2: Offset volume descriptor.

```
00000000  33 C0 FA 8E  D8 8E D0 BC  00 7C 89 E6  06 57 8E C0  3........|...W..
00000010  FB FC BF 00  06 B9 00 01  F3 A5 EA 1F  06 00 00 52  ...............R
00000020  52 B4 41 BB  AA 55 31 C9  30 F6 F9 CD  13 72 13 81  R.A..U1.0....r..
00000030  FB 55 AA 75  0D D1 E9 73  09 66 C7 06  8D 06 B4 42  .U.u...s.f.....B
00000040  EB 15 5A B4  08 CD 13 83  E1 3F 51 0F  B6 C6 40 F7  ..Z......?Q...@.
00000050  E1 52 50 66  31 C0 66 99  E8 66 00 E8  35 01 4D 69  .RPf1.f..f..5.Mi
00000060  73 73 69 6E  67 20 6F 70  65 72 61 74  69 6E 67 20  ssing operating
00000070  73 79 73 74  65 6D 2E 0D  0A 66 60 66  31 D2 BB 00  system...f`f1...
00000080  7C 66 52 66  50 06 53 6A  01 6A 10 89  E6 66 F7 36  |fRfP.Sj.j...f.6
00000090  F4 7B C0 E4  06 88 E1 88  C5 92 F6 36  F8 7B 88 C6  .{.........6.{..
000000A0  08 E1 41 B8  01 02 8A 16  FA 7B CD 13  8D 64 10 66  ..A......{...d.f
000000B0  61 C3 E8 C4  FF BE BE 7D  BF BE 07 B9  20 00 F3 A5  a......}.... ...
000000C0  C3 66 60 89  E5 BB BE 07  B9 04 00 31  C0 53 51 F6  .f`........1.SQ.
000000D0  07 80 74 03  40 89 DE 83  C3 10 E2 F3  48 74 5B 79  ..t.@.......Ht[y
000000E0  39 59 5B 8A  47 04 3C 0F  74 06 24 7F  3C 05 75 22  9Y[.G.<.t.$.<.u"
000000F0  66 8B 47 08  66 8B 56 14  66 01 D0 66  21 D2 75 03  f.G.f.V.f..f!.u.
00000100  66 89 C2 E8  AC FF 72 03  E8 B6 FF 66  8B 46 1C E8  f.....r....f.F..
00000110  A0 FF 83 C3  10 E2 CC 66  61 C3 E8 76  00 4D 75 6C  .......fa..v.Mul
00000120  74 69 70 6C  65 20 61 63  74 69 76 65  20 70 61 72  tiple active par
00000130  74 69 74 69  6F 6E 73 2E  0D 0A 66 8B  44 08 66 03  titions...f.D.f.
00000140  46 1C 66 89  44 08 E8 30  FF 72 27 66  81 3E 00 7C  F.f.D..0.r'f.>.|
00000150  58 46 53 42  75 09 66 83  C0 04 E8 1C  FF 72 13 81  XFSBu.f......r..
00000160  3E FE 7D 55  AA 0F 85 F2  FE BC FA 7B  5A 5F 07 FA  >.}U.......{Z_..
00000170  FF E4 E8 1E  00 4F 70 65  72 61 74 69  6E 67 20 73  .....Operating s
00000180  79 73 74 65  6D 20 6C 6F  61 64 20 65  72 72 6F 72  ystem load error
00000190  2E 0D 0A 5E  AC B4 0E 8A  3E 62 04 B3  07 CD 10 3C  ...^....>b.....<
000001A0  0A 75 F1 CD  18 F4 EB FD  00 00 00 00  00 00 00 00  .u..............
000001B0  00 00 00 00  00 00 00 00  AF 06 E1 FB  00 00 80 20  ...............
000001C0  21 00 0C AB  81 6A 00 08  00 00 00 A0  97 00 00 AB  !....j..........
000001D0  82 6A 83 5B  F9 E2 00 A8  97 00 00 18  5C 00 00 00  .j.[........\...
000001E0  00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000001F0  00 00 00 00  00 00 00 00  00 00 00 00  00 00 55 AA  ..............U.
```

Figure 3: Booting with MBR from a flash disk.

LISTING 1: Creating the Knoppix Image

```
xorriso -compliance iso_9660_level=3 -as mkisofs -l -r -J -iso-level 3
  -isohybrid-mbr /usr/lib/ISOLINUX/isohdpfx.bin -V "KNOPPIX_8" -A "KNOPPIX V8.0"
  -no-emul-boot -boot-load-size 4 -boot-info-table -b boot/isolinux/isolinux.bin
  -c boot/isolinux/boot.cat -hide-rr-moved -eltorito-alt-boot -e efi.img
  -no-emul-boot -append_partition 3 0x83 REISERFS.IMG -isohybrid-gpt-basdat
  -o KNOPPIX8_IMAGE.iso KNOPPIX8_DIRECTORY
```

sizes, ended by a `0x55AA` signature (which may be used by the BIOS to check if this really looks like a standard partition table).

The partition table at offset `1BE` results in the following partitioning shown with `fdisk -l` (Figure 4).

Merging ISO 9660 El Torito and MBR Booting

As mentioned before, a USB flash disk will usually not become bootable by just copying the DVD's ISO image over it. The most apparent solution is adding an MBR with a boot code and a partition table in the first 512 bytes of the image, which are not used by the ISO 9660 filesystem anyway. The result is a hybrid (El Torito + MBR) booting system that can be burned to DVD or flashed to a disk with no changes.

For Knoppix, the hybrid image partition table (Figure 5), which is injected into the first sector of the image, contains three partitions. The first "empty" one is the original ISO 9660 filesystem; the other two will be explained in the next sections.

The boot code is created from the isolinux hybrid boot loader, which comes as `isohdpfx.bin` with isolinux. For actually creating the image, I use `xorriso`.

After many experiments, the final command line for creating the final Knoppix image can be seen in Listing 1.

In theory, the steps of inserting additional information into the image could have been done with `dd` and slices of binaries, but using `xorriso` is just more convenient.

Most Linux distros already use this method in their installers and live images in one way or the other. However, there are some disadvantages in this approach of an "all-in-one" image, which made me hesitate to switch to the hybrid image format until now:

- When flashing to disk instead of burning a DVD, the first partition containing the ISO 9660 filesystem will be read-only, simply because ISO 9660 does not support any write operations regardless of writeability of the installed medium.
- Some BIOSes get confused when they see a partition table (or anything but zero bytes) in the first sector of a DVD and refuse to even display it in the boot menu.
- The additional partition table in the first sector may be interpreted with the sector size of 2048 bytes instead of the standard 512 bytes of the DVD when trying to just mount it from a drive.
- The ISO 9660 filesystem starts at sector `0` (which is the same sector where the MBR and partition table are located). When inserting a partition entry for mounting the ISO, this also may confuse some software that assumes that a partition can only start at sectors higher than `0`. In fact, the ISO filesystem really overlaps the MBR and partition table, which are embedded in the unused first blocks of the filesystem.
- Since the partition offsets and sizes specified in the partition table have to be set to a fixed number, they will

```
Disk /dev/sdb: 7.6 GiB, 8178892800 bytes, 15974400 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xfbe106af

Device     Boot   Start      End Sectors  Size Id Type
/dev/sdb1  *       2048  9938943 9936896  4.8G  c W95 FAT32 (LBA)
/dev/sdb2       9938944 15974399 6035456  2.9G 83 Linux
```

Figure 4: Partition table at offset `1BE`.

```
Disk KNOPPIX_V8.0-LM-Beta5.iso: 4.2 GiB, 4547162112 bytes, 8881176 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x6a153c55

Device                      Boot   Start     End Sectors  Size Id Type
KNOPPIX_V8.0-LM-Beta5.iso1 *           0 8872919 8872920  4.2G  0 Empty
KNOPPIX_V8.0-LM-Beta5.iso2         28808   59527   30720   15M ef EFI (FAT-12/16/32)
KNOPPIX_V8.0-LM-Beta5.iso3       8872920 8881111    8192    4M 83 Linux
```

Figure 5: The hybrid image partition table for Knoppix contains three partitions.

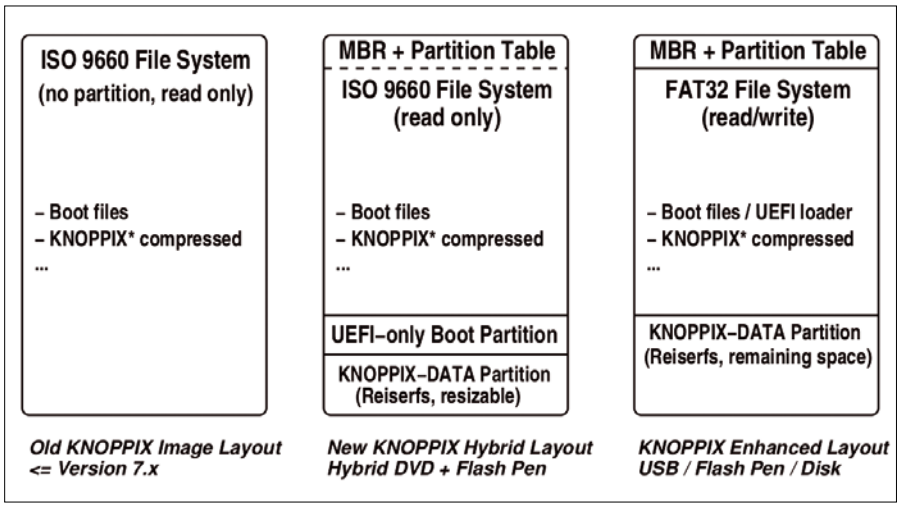


Figure 6: Image layouts for three versions of Knoppix (old, hybrid, and enhanced).

most likely *never* exactly match the physical size of the target disk, and they are also not aligned to "cylinders" (which causes warnings in some partitioning programs).

Consequently, some computers will no longer boot the DVD after switching to hybrid boot format.

Compatibility?

The big question for me was (and still is): How many computers are affected by not being able to boot the new format. Sure, refusing to boot a DVD that still follows the valid standard for El Torito booting can be considered a bug in the BIOS (aka, "It's not my fault…").

However, I'd rather renounce new features than lose the capability to boot on as many different computers as possible. Also, it is not transparent to users why a Knoppix 7 DVD booted without problems, while version 8 won't even show up in the boot menu anymore, in those (hopefully) rare cases.

On the other hand, the hybrid format finally allows distributing an image that works (on most computers) for DVDs, as well as USB flash disks, by just writing the same data. It seems to me that it's unavoidable in the long run because of the disappearance of DVD drives from modern computers.

Improving Bootability

Knoppix includes a tiny "boot-only" ISO image in the KNOPPIX folder, which allows you to boot computers that have a USB but cannot boot from it because of a BIOS limitation, to first boot kernel and initial ramdisk from the tiny (20MB) CD version. This then searches for the KNOPPIX folder in all attached drives containing the main system files. This also may be a good workaround for non-hybrid booting BIOSes.

For a UEFI boot, I included a FAT32 partition image with a signed 64-bit pre-bootloader, which then chainloads a special EFI version of syslinux (the disk bootloader) for starting kernel and initial ramdisk in this boot mode.

In theory, computers that can only boot via UEFI should find this image as a UEFI partition on the DVD, as well as on flash disk, and boot from there. In UEFI "secure" mode, a dialog will ask for permission to enroll the bootloader's checksum in the UEFI firmware to enable execution of the "foreign" OS.

With secure mode turned off, UEFI will start the boot screen normally (just without the usual boot splash graphics; we are not in VGA mode anymore).

Image Layout

Figure 6 shows the old DVD-only layout on the left, the new hybrid DVD + Flash layout in the middle, and the "improved" format on the right which allows you to write on the first partition again. The latter is still created by the included `flash-knoppix` and is not usable for DVDs.

Only the enhanced layout allows to use special Knoppix features, such as changing boot options (the first partition needs to be writable for that) and encryption of the data overlay on the last partition.

Regaining Unclaimed Space

You may have noted the tiny (4MB) Linux partition at the end of the hybrid image format. This is a trick that allows me to adjust the last partition automatically with no interaction during the first boot of a USB flash disk that was freshly installed from the hybrid image using `dd` or Disk Imager.

During the first boot of Knoppix in the hybrid layout, the idea is to:

1. Expand the last (`KNOPPIX-DATA`) partition of the disk to the maximum size.
2. Expand the filesystem (ReiserFS) on the partition to also use the now available full capacity.
3. Continue booting normally with the adjusted `KNOPPIX-DATA` partition as a writable overlay for personal data.

As seen with other distros, there is usually a reboot issued after modifying partitions. I'm trying to avoid this by letting the Knoppix ramdisk do the adjustment before actually mounting the partition, so the kernel can reread the new partition table without problems.

Even if rereading the partition table fails (you never know…), the data partition's start is unchanged, because it was already there, so it can already be used regardless of a temporary "wrong" partition size. Somewhat to my surprise, this works perfectly on all tested computers, so you have a full-sized data partition for your installations and configuration changes immediately from the first boot.

The relevant section in the initial ramdisk script makes sure that *only* a partition labeled as `KNOPPIX-DATA` will be expanded, so no other partitions are affected accidentally.

Conclusion

Switching to hybrid format for the distribution of Knoppix makes it possible to skip the step of creating DVDs. With a few caveats, compatibility with reluctant firmware can be circumvented.

The non-hybrid enhanced format can still be created by `flash-knoppix` in order to create a fully writable system with optional encryption. ■■■

INFO

[1] El Torito Specification: *http://download.intel.com/support/motherboards/desktop/sb/specscdrom.pdf*

Create your own Linux distribution

MASTER BUILDER

Linux systems out of the box are a dime a dozen. But if you want a customized system, you will certainly see the benefit of AryaLinux, which lets you put together an individualized system from the sources. *By Erik Bärwaldt*

Several hundred different Linux distributions for different use cases vie for your favor. These preconfigured systems also include some software that you find useless, or they lack important programs that you need to install manually. AryaLinux from India takes a different approach here: The operating system is developed from Linux From Scratch (LFS; i.e., from source) and targets advanced individuals who prefer to determine what is installed on their system and want to curate their own software.

Images

Although the project site now offers 2017 images, for this article, I used the 2016 images, which are still available for 32- and 64-bit architectures [1]. The hybrid images also allow you to start up from a USB stick. From 2017 on, AryaLinux will only be offered in the 64-bit variety.

The 2016.08 images come in a choice of two desktop environments: In addition to the lean Mate interface, whose

ISO image weighs in at roughly 2.3GB, you can get the not-so-lean Xfce desktop (2.7GB) [2]. The developers also provide a Builder DVD for both popular hardware architectures, which is fairly small at 1.6GB but is based on the older 2016.04 version. The 2017 images are significantly smaller at 1.9GB (Mate), 1.8GB (Xfce), and 1.4GB (Builder).

Live System

After creating a bootable storage medium, the Live system boots to a visually appealing desktop that immediately displays a full-screen window, without any intervention by the user, for localizing the system. You can select the language option from the numerous alternatives. After then clicking *Change Language*, the desktop will appear in your choice of language within a few seconds.

Besides the traditional icons for file management, the interface also offers a starter for installing the operating system. In a style familiar from the older Gnome 2.x branches, you will also find two horizontal panel bars at the top and bottom edges of the screen, with the Applications, Places, and System menus on the left and a system tray on the right to facilitate the use of the system.

AryaLinux does not seem unusual when you look at the individual submenus: You will find the entire spectrum of standard applications, including LibreOffice, Gimp, Firefox, and Thunderbird. In addition to some smaller Mate-specific applications, the software inventory includes some less frequently preinstalled applications, such as the universal VLC media player or the graphical mass memory management tool GParted.

When you call individual programs, full localization is a positive aspect: AryaLinux also translates external applications (e.g., Gimp, VLC, and Xfburn) from the Xfce treasure trove. The system is thus fine for immediate use without any further manual adjustments.

Installation

To install AryaLinux on a mass storage device, use the *AryaLinux Installer* launcher on the desktop. It takes you to a routine that sets up the system on the hard drive in six steps. Somewhat unusual is the way the system sets up the partitions compared with other installers. The wizard is easy to use with its self-explanatory settings.

AryaLinux needs at least 20GB of local storage for the system partition, as well as (for 2G of RAM or less) a swap partition of at least 2GB, which also is used for hibernation mode. Because the wizard itself does not rely on partitioning software, it is advisable to prepare the local mass storage first by using the GParted built-in tool in Live mode. In the following two dialogs, you specify a user and set an administrator password.

The localization options appear next and are divided into three parts: In addition to the system language, the routine asks you for the desired keyboard layout and time zone. After you have approved the configuration, the installer installs the system on your mass storage (Figure 1). After a quick restart, AryaLinux is then available; the software selection matches that of the Live system.

Adaptable

The AryaLinux derivative lets you customize the desktop. A variety of tools is available in the *System* | *Control Center* (Figure 2). Other tools, such as the *Fusilli Settings Manager* or *Rotini Theme Manager*, located under *System* | *Preferences* | *Look and Feel*, offer a variety of options for customizing the desktop.

Additional Software

If you want to include additional applications in the system, several hundred applications are available for installation. These cannot be integrated easily into the system using a graphical tool

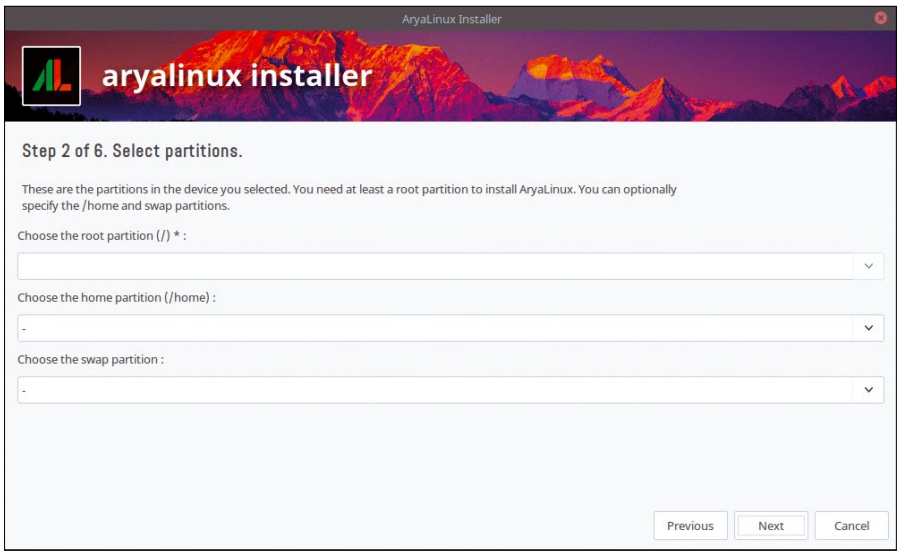


Figure 1: The installer puts the system on your mass storage in just a few steps.

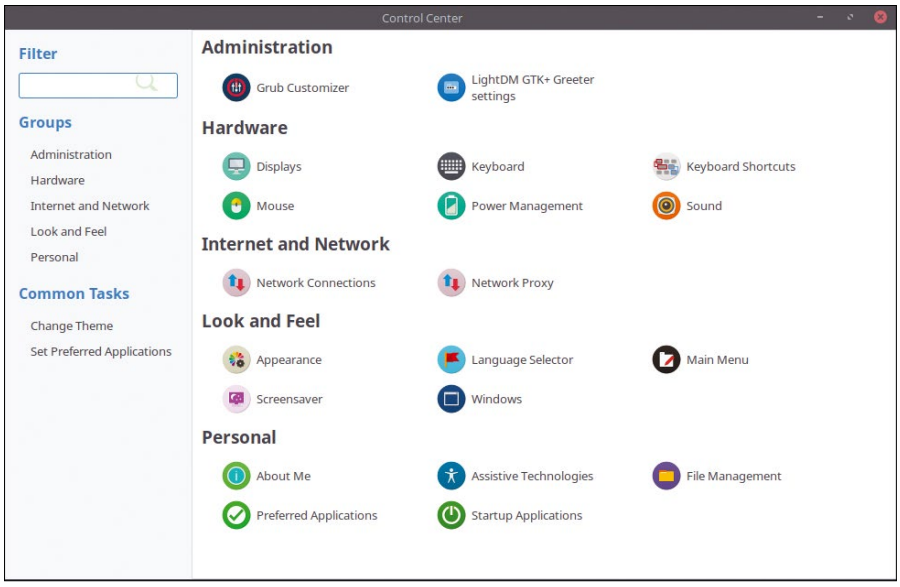


Figure 2: The Control Center combines all the important settings.

but have to be built manually. The AryaLinux Packaging System (ALPS) is used in this case.

Although ALPS resolves dependencies, it is not a full package management system like RPM or DEB; it is primarily intended to facilitate the task of creating packages from the sources, which explains why there is no graphical front end, like Synaptic or YaST. Entering the `alps help` command at the prompt shows the tool's capabilities in an overview.

Working with administrator privileges, you can add new applications to the system with the

```
alps install <program>
```

command. To install the Java 7 Runtime Environment, for example, just type:

```
alps install java7
```

This prompts ALPS to integrate the corresponding OpenJDK version.

The installation takes significantly longer than with pre-built RPM or DEB packages because of the programs that need to be compiled. You can discover which applications can be installed by checking out the `/var/cache/alps/scripts/` directory: The system lists all programs available in the form of installation scripts.

Chain Reaction

AryaLinux's unique selling point is the option for creating an individualized operating system. Other distributions (e.g., PCLinuxOS) implement such an option by customizing the existing system; however, the new operating system is created completely from the sources in AryaLinux using pre-built scripts, which means the system has no unnecessary "ballast" that could affect the working speed. AryaLinux offers a chain of scripts (the "toolchain") that lead to predefined workflows. The chain ends with the complete operating system.

Step by Step

To build in individualized AryaLinux, download the Builder DVD from the project site and transfer the image to an optical disc or a USB flash drive; then, start the operating system from the medium.

If not already automatically enabled, set up an Internet connection, because AryaLinux first updates the existing scripts. Next, switch to a terminal and type the following command sequence:

```
$ sudo su
# cd
# cd scripts
# ./1.sh
```

The routine now updates the existing scripts and asks about localization requirements. You can choose from English or Italian. The software lists the existing mass storage devices and prompts for the name of the device on which you want to install the build system.

The routine displays the partitions on this disk. You need to partition the corresponding mass storage device with a suitable partition table before you launch AryaLinux, because the system itself has no tools to accomplish the task. Now the software prompts you to identify the root, home, and swap partitions for the build system before asking you to define a hostname and the basic credentials for a user account.

In the next step, you define the locale. For U.S. English, for example, you enter *en_US. utf8* followed by your choice of paper size (e.g., *A4*), time zone, and the corresponding country; the routine provides preconfigured lists. After this dialog, the tool prepares the target partitions accordingly.

Calling the `./2.sh` script and running the `exit` command, temporarily changes the root directory, builds the toolchain, and populates the target partition with some essential files and directories. Next, enter the `resume` command at the prompt, and start the third script with the command `./3.sh` when prompted to do so. After entering `exit` twice, you finally end the subprocesses and seal off the root directory.

To create the kernel, change to the source directory (`cd /sources`) and call `./4.sh`, the fourth script, which creates the kernel. You are asked for a root password and some personal details relating to the administrator. The routine then loads and installs more system software. Exit the script again, when prompted to do so, by typing `exit`.

The last step for the routine is to configure the bootloader. If you see no selection option for the new system in the GRUB menu on reboot, allow the system to start up and then configure the bootloader manually by entering the command

```
# grub-mkconfig -o /boot/grub/grub.cfg
```

at the prompt. You can now complete and tweak the new system as desired.

Additions

You can add a graphical user interface to any customized system by generating it from the source, so you should use the Builder DVD to set up a desktop environment, as well.

The developers provide detailed instructions for integrating Plasma 5 or the lightweight LXQt desktop online [3], where they also explain how you can build a complete LAMP system as the basis for other services.

Scripts for additional desktop environments are under development but not yet operational. Thus, although you can call several scripts to install the LXDE, LXQt, and Gnome desktops in the prefabricated AryaLinux variants, they all produce error messages because of missing dependencies.

If you need assistance with various problems, you can turn to the AryaLinux Forum [4], which the developers conscientiously support and which enjoys great popularity.

Conclusions

The Builder version of AryaLinux is a great resource for advanced users who want to install an operating system from the sources themselves. Users who need only a few standard applications and are looking for a lean, attractive desktop should also take a look at the prefabricated AryaLinux versions. Because the system is still very young and experimental, it does fail in some areas, but dedicated users can make a valuable contribution by adding their knowledge and experiences to the development work. ■■■

INFO

[1] AryaLinux project site: *http://aryalinux.org*

[2] Download: *http://aryalinux.org/downloads/*

[3] Guides: *http://aryalinux.org/docs/*

[4] Forum: *http://www.linuxquestions.org/questions/aryalinux-120/*

Your Roadmap to the Open Hardware Revolution ...

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

Order your subscription today and tune in to the revolution!

shop.linuxnewmedia.com

Print Sub

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

Digital Sub

Our PDF edition is a convenient option for mobile readers.

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

$59.95 £37.50 €44.90

Download on the App Store

GET IT ON Google play

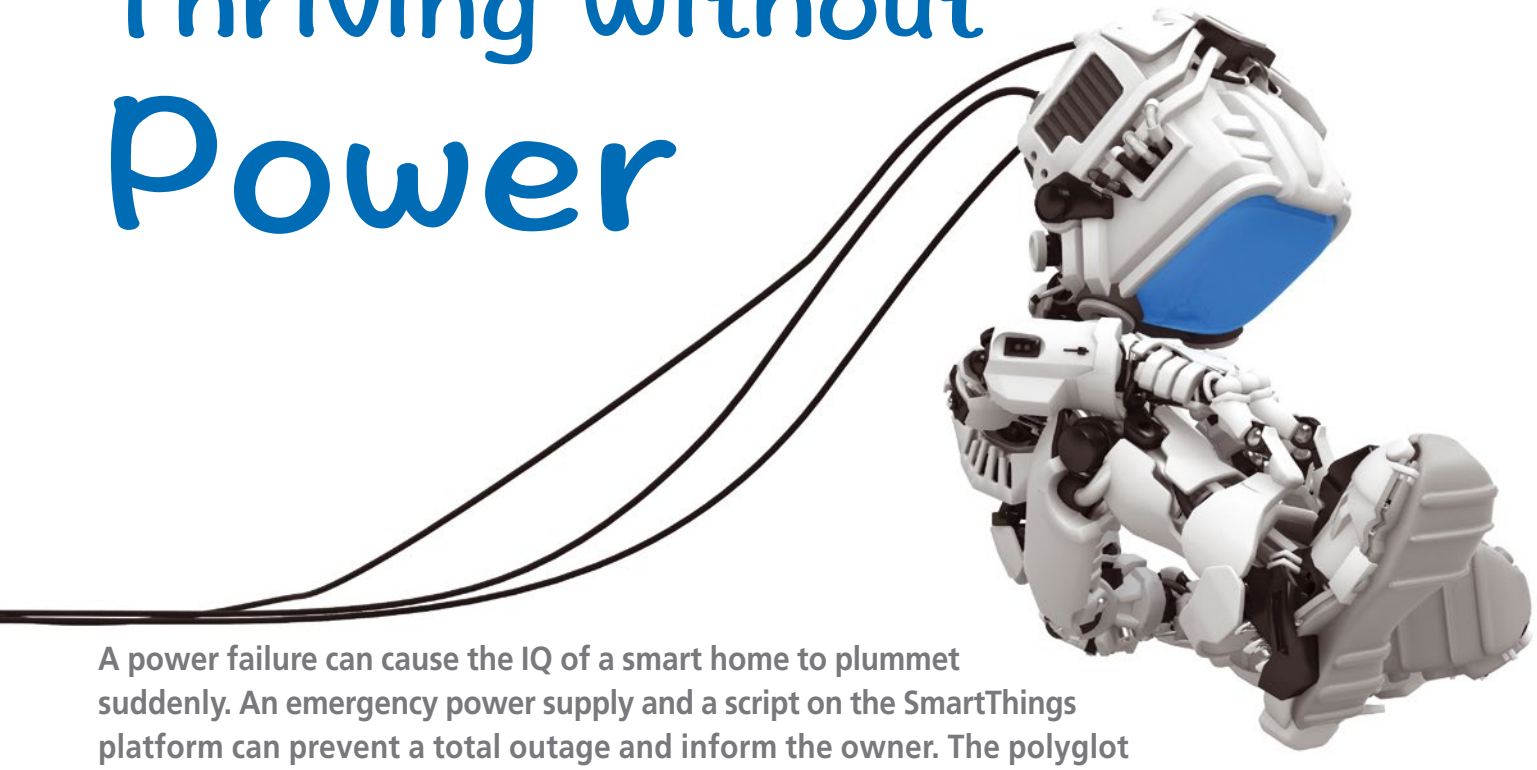
iTunes Store

Google US

Google UK

Mastering a power outage in the smart home

Thriving Without Power



A power failure can cause the IQ of a smart home to plummet suddenly. An emergency power supply and a script on the SmartThings platform can prevent a total outage and inform the owner. The polyglot Perlmeister embarks on a foray into the territory of the Groovy scripting language in this issue. *By Mike Schilli*

Hard to believe but a sad reality: Even in major cities in the United States, power outages from one second to the next are not uncommon, whether because of the wild mess of overhead cables stretching between houses in the City of San Francisco (Figure 1) or because the electrical infrastructure of the substations is outdated and starting to crack under the strain. You should always expect the worst as a smart homeowner and implement strategies, just in case the power stops flowing for a few hours. Even if you live in a country with a more stable power supply, you will still appreciate the fail-safes introduced here. Then, if a shrewd burglar takes the precaution of removing a fuse before invading your privacy, they will definitely get a good scare when the alarm still goes off!

MIKE SCHILLI

Mike Schilli works as a software engineer in the San Francisco Bay area of California. In his column, launched back in 1997, he focuses on short projects in Perl and various other languages. You can contact Mike at *mschilli@perlmeister.com*.

Emergency Power

In case of an outage, I want my key automation components, such as the controlling hub and its connection to the Internet – including the path thereto encompassing the router and cable or DSL modem – to continue working without electricity. If you limit yourself to keeping just a few essential components alive during a power outage, the consumption adds up to just a few watts; even a cheap battery-backed emergency power supply for less than $50 can keep this minimal infrastructure alive for a while.

Choosing the Hardware

Devices known as uninterruptible power supplies (UPSs) typically specify their characteristics in volt-amperes (VA), which is unfortunately not sufficiently helpful to answer the most fundamental question: How long will they keep a consumer with a known power consumption rate running? To find out, you need to know more about the fundamentals of the internal batteries and

Figure 1: San Francisco: Supplying power through a bird's-nest of cables can lead to failure.

Figure 2: The SmartThings hub (top) and the emergency power supply for the Internet connection.

plug this information into a formula [1] that will give you the number of minutes to the point of exhaustion.

Initially, I bought a cheap UPS for $30, but it kept bugging me with senseless beeping during power outages and wouldn't even let me disable it, because, apparently, it knew better. Finally, I grudgingly plunked down $120 for a solidly crafted UPS by Tripp-Lite (Figure 2), which digitally indicates how many minutes of juice are left for battery-backed operation and has a "quiet" key, which ensures that the device does its job without beeping during a power outage. The wonders of technology!

Smart Home Despite Outage

As mentioned in a previous article [2], the SmartThings home automation kit I set up includes a controlling hub and a set of four different sensors and actuators; until recently, it was only available in the United States. In September 2015, a customized version of the SmartThings hub starter kit [3] for the UK was released for the European market; you can order it on Amazon for about $199 (EUR215/£199). Additional suitable sensors and actuators that support the Z-Wave or ZigBee Protocol are available for about $40 (EUR40/£30).

Most of the components in this smart home solution are battery operated; even the controller is battery-buffered, and when the mains voltage goes missing,

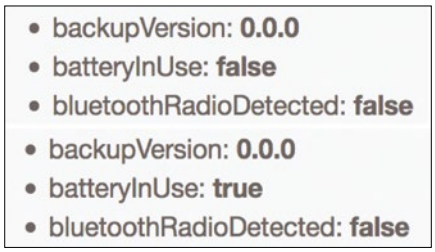


Figure 3: The hub indicates whether it is running on mains voltage or on battery power during an outage.

no further measures are needed to switch seamlessly to battery. Also, my Arlo wireless surveillance cameras [4] run on battery power, so the UPS only needs to back up the Arlo base station, the router, and the ISP modem. Virtually everything continues running for up to three hours with this simple solution if the power fails. Outages of more than an hour are thankfully rare, even out here in the Wild Wild West.

Construct

But how does a sensor discover that the power line is down and the emergency generator has started up? It would be obvious to ask the controller itself, and in fact its web interface (Figure 3) shows whether it is running on emergency battery power or if the mains voltage is available. Unfortunately, SmartThings does not offer this information in the developer API [5].

I thus dreamed up a solution inspired by the master of complicated technical solutions, Rube Goldberg: What about an opaque distribution box from the hardware store, into which I route a power cable to power an LED nightlight inside consuming only 0.3W, as well as a battery-powered light sensor that checks

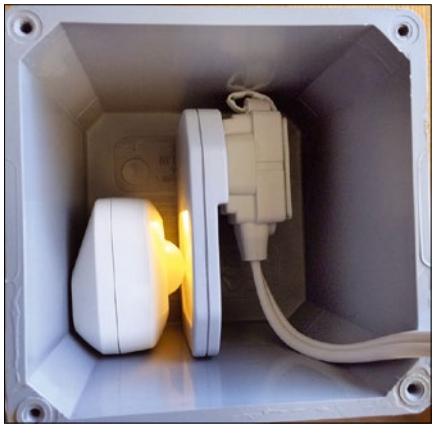


Figure 4: An LED nightlight is lit as long as the power is on, and the sensor indicates that the juice is flowing.

whether the light, and thus the power, is on (Figure 4)? My choice of sensor was a device known as a Zooz (Figure 5), which acts as a motion detector and measures the room temperature and light exposure in lux.

The SmartThings hub pretty much integrates every device that speaks the wireless Z-Wave or ZigBee protocols; thus, integrating the sensor with the hub in the mobile phone app was easy as pie (Figure 6).

SmartApps

The SmartThings app for the mobile phone (iOS and Android) reads the sensor

Figure 5: The battery-powered light-motion-temperature sensor by Zooz speaks Z-Wave with the controller.

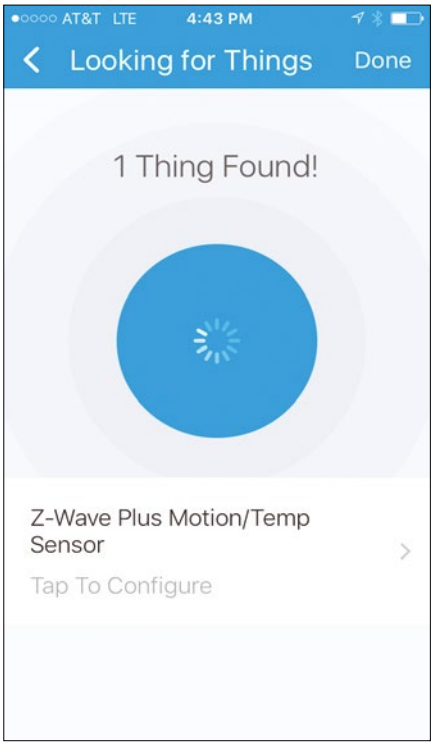


Figure 6: The hub often finds new Z-Wave components automatically in inclusion mode.

data at regular intervals or simply uses a subscription to be notified when predefined events occur. The Dashboard overview (Figure 7) shows the current state of all devices. Switches can also be flipped using an app – and the app displays their final state.

Any needs beyond this must be programmed by the user. Developers can paste together Groovy code to create so-called SmartApps, which query sensors, initiate actions with Z-Wave actuators, or fire off external web requests. SmartThings is tight-lipped when users ask where these SmartApps actually run: on the hub or in the cloud? Depending on the load, the company reserves the right to perform the necessary computational steps here or there.

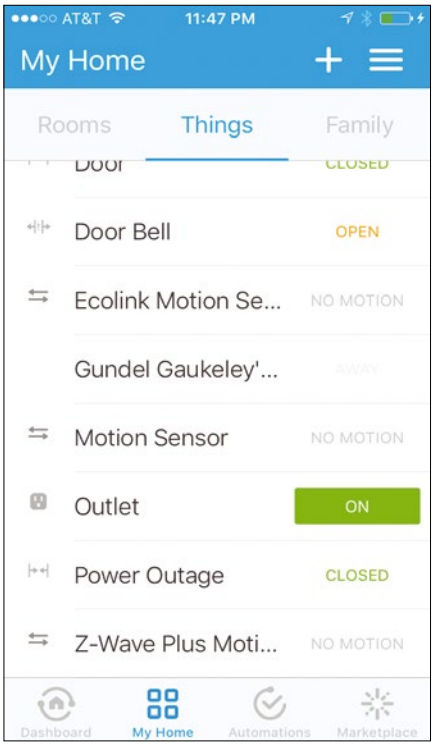


Figure 7: The SmartThings app displays the state of all devices and controls them individually.

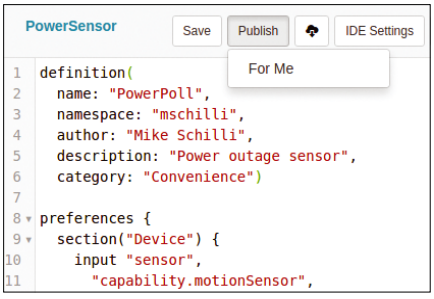


Figure 8: Users can write their own applications for the SmartThings hub in an IDE in Groovy and then install them by pressing *Publish*.

A simulator in the IDE helps eradicate potential final teething troubles, and when you press *Publish* | *For Me* (Figure 8), the phone app installs the code previously edited in the desktop browser on the hub. Magic!

The `preferences` section of the Groovy code in Listing 1 (lines 8-14) [6] narrows down the selection of sensors from which the user must choose after launching the newly installed SmartApp in *Marketplace* | *My Apps*. The script only expresses interest in devices with a `motionSensor` capability, and the SmartThings app queries the hub for all devices with this property and presents a list of choices to the user.

If you select the newly added Zooz Z-Wave Plus Motion/Temp Sensor, the SmartApp launches, and the stunned developer can track the log output of the app running somewhere in the cloud in an any browser window (Figure 9).

LISTING 1: zooz.groovy

```
01 definition(
02   name: "PowerPoll",
03   namespace: "mschilli",
04   author: "Mike Schilli",
05   description: "Power outage sensor",
06   category: "Convenience")
07
08 preferences {
09   section("Device") {
10     input "sensor",
11       "capability.motionSensor",
12       required: true
13   }
14 }
15
16 def installed() {
17   initialize()
18 }
19
20 def updated() {
21   initialize()
22 }
23
24 def initialize() {
25   unschedule()
26   schedule("42 * * * * ?", handler)
27 }
28
29 def handler() {
30   log.debug "Light Check: " +
31     sensor.currentIlluminance
32 }
```

Fixed Events

The code itself is event driven; the functions `installed()` and `updated()` are required entry points that the hub jumps to after the user reinstalls or updates the SmartApp. Listing 1 funnels both events into the `initialize()` function in lines 24-27, which creates a cron entry that jumps to the `handler()` function defined in line 29 every 42nd second of each minute. Before creating the new cron entry, `unschedule()` deletes all previously created entries, just in case, to avoid an avalanche of new entries on frequent installs or updates.

The `currentIlluminance` method of the sensor object in line 31 reads the light exposure value from the sensor, whose name it picked up previously in line 10. The fact that a string in line 10 generates an object in line 31 without so much as a by-your-leave is what is known in professional circles as "spooky action at a distance." Experienced programmers fear this effect like the devil fears holy water, but the SmartThings developer API is full of such folly, unfortunately.

As the log output in Figure 9 shows, it turns out that the hub only reads out the Zooz sensor every few minutes. Also the Zooz device does not seem to support a subscription mode offered by other devices, in which the hub immediately jumps to a callback when a sensor value changes.

When power fails, it can thus take five minutes until the code notices the event and therefore can't initiate actions immediately, such as notifying users via text message. This is not the end of the world, but there is a better way.

Another Sensor

On Amazon, I found another Z-Wave sensor by seven7express (Figure 10) that reports power outages very efficiently. It is implemented as a door sensor and reports `closed` if the power is on and `open` if not. It is connected to a power outlet via a charger/adapter, whereas the sensor itself has an internal battery that allows it to send signals to the hub via Z-Wave if the wall socket stops supplying power.

The device worked right off the bat. The Groovy script in Listing 2 implements the logic for controlling it. In the `preferences` section, the hub looks for sensors

with the `capability.contactSensor` property and offers the user the newly installed power sensor, among other things, to select from when you launch the app. Line 26 obtains a subscription to the `contact` event, which the hub triggers each time the sensor transitions from `open` to `closed` and vice versa. The script thus notices power outages with only one- or two-second delays and logs them in lines 31 and 33.

Hello User!

Once the SmartApp determines that the power has gone, it uses the Prowl Web API, as described in a previous article [7], to notify the user who has installed the Prowl app on their cell phone. Listing 3 plants the API key required by Prowl, which registered users can pick up from the website [8] after purchasing the app for $3.

The `prowl()` function expects a message (`Power outage!` or `Power back!`) passed in, sets the event field with it, and adds the name of the sending app and a short explanation so the receiver knows where the message came from. In the `try` block, `httpGet` then issues the web request and checks whether the outcome is good or bad, and `log.debug` logs the message for later error analysis. Prowl is a simple and efficient service that works both on iOS and Android (Figure 11).

Web Access

If you enable the slightly hidden OAuth section when setting up the SmartApp on the SmartThings website, you will receive an OAuth token after an OAuth token dance (e.g., via the CPAN OAuth::Cmdline module;

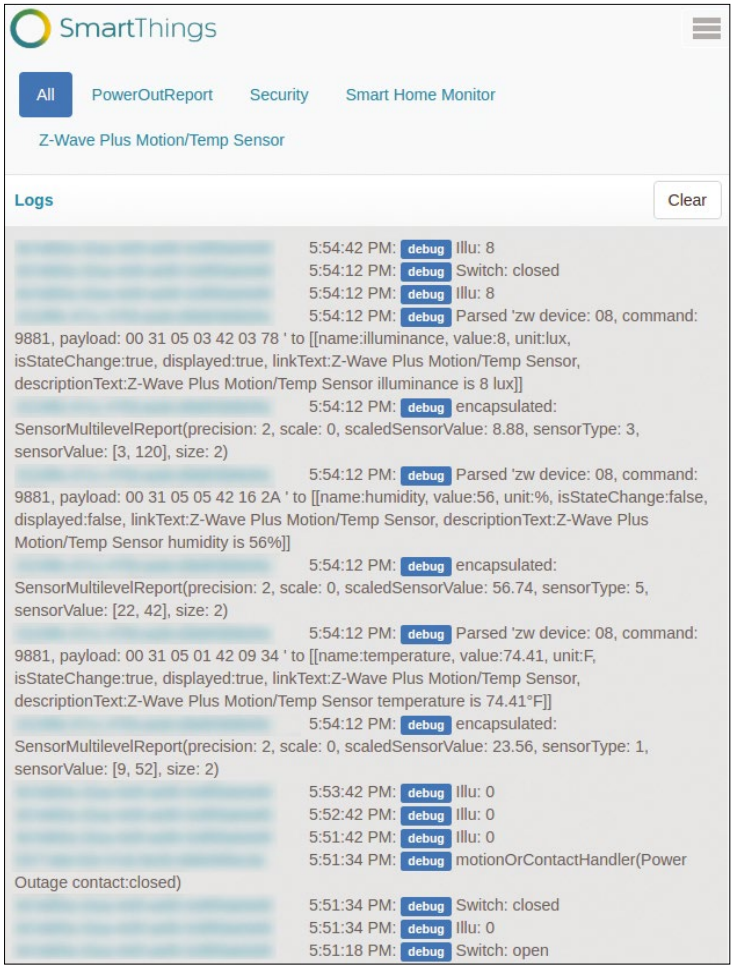


Figure 9: The hub queries the light sensor every four minutes, whereas events on the contact sensor appear immediately.

LISTING 2: power-sensor.groovy

```
01 definition(
02   name: "PowerSensor",
03   namespace: "mschilli",
04   author: "Mike Schilli",
05   description: "Subscribe and detect",
06   category: "Convenience")
07
08 preferences {
09   section("Device") {
10     input "power",
11       "capability.contactSensor",
12       required: true
13   }
14 }
15
16 def installed() {
17   initialize()
18 }
19
20 def updated() {
21   initialize()
22 }
23
24 def initialize() {
25   unsubscribe()
26   subscribe(power, "contact", evthandler)
27 }
28
29 def evthandler(evt) {
30   if(power.currentContact == "closed") {
31     log.debug "Power back!"
32   } else {
33     log.debug "Power outage!"
34   }
35 }
```

Figure 10: The battery-powered power outage sensor by seven7express shows the user whether or not power is available as a door sensor.

LISTING 3: prowl.groovy

```
01 def prowl(message) {
02   def params = [
03     uri: "https://api.prowlapp.com",
04     path: "/publicapi/add",
05     query: [
06       event: message,
07       application: "Smartthings",
08       url: "",
09       description: "Power Outage Notification",
10       apikey: "xxxxxxxxxxxxxxxxxxxxxx"
11     ]
12   ]
13
14   try {
15     httpGet(params) { resp ->
16       log.debug "Prowl message sent";
17     }
18   } catch(e) {
19     log.error "Can't send Prowl message: $e"
20   }
21 }
```

LISTING 4: webapi.groovy

```
01 mappings {
02   path("/power") {
03     action: [
04       GET: "checkPower"
05     ]
06   }
07 }
08
09 def checkPower() {
10   return [power:
11     power.currentContact == "closed" ?
12       "ok" : "not ok"]
13 }
```

Figure 12); you can then access the running SmartApp from the Internet via a web API. The `mappings` section of the SmartApp code in Listing 4 defines the entry points of the web API and the actions assigned to them. The `checkPower()` function defined in lines 9-13 then reads the current state of the sensor on demand and returns the result as a map, which the web API then returns to the web client in JSON format.

Figure 13 shows the query using a `curl` client at the command line. First, it asks for the location of the `endpoints` stating the API token (i.e., the complete URL under which a registered user can find their SmartApps). Armed with this URL, the client can access the entry points defined in the code (in this example, `/power`) and receive JSON-formatted output in return (e.g., the value in the `power` field is `ok` or `not ok`).

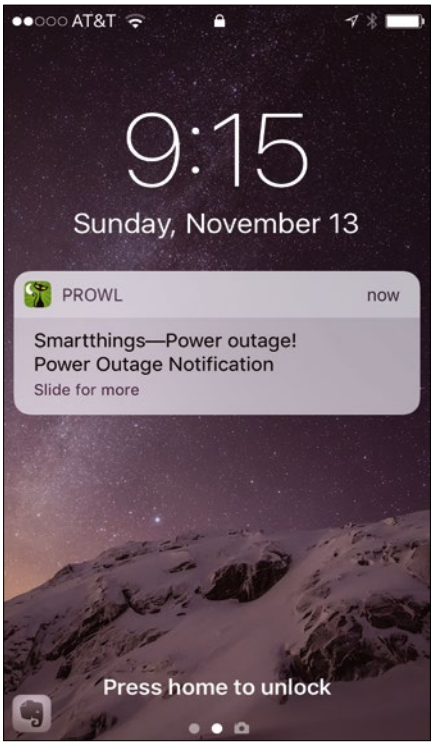


Figure 11: The text message tells the user about a power outage at home.

Bad Condition

Programming with the SmartThings API is recommended only for hobbyists with a great deal of patience. Samsung's SmartThings department seems to be the unloved stepchild of the organization's other divisions. Not only is the documentation inadequate and at times contradictory, even the source code of the hub core is inaccessible. Even worse, Groovy's own self-inspection mechanisms have been undermined so that not even go-getters can discover the true names of incorrectly documented attributes.

In the case at hand, a simple API call to ask the hub whether it currently is powered by the grid or the reserve batteries would have sufficed to solve the problem, instead of having to resort to Rube Goldberg-esque solutions such as the one I just described. If the hub code were open, dedicated hobbyists could fill in the gaps in the API on GitHub and make it a much better product.

The app keeps crashing, the web page for developing SmartApps has been cobbled together rather haphazardly, and serious functional flaws are evident (e.g., the *Update* button for changing a SmartApp's metadata does not work). If you think that developer access will work via *graph.api.smartthings.com*, you will discover in the forum that you need to stipulate *graph-na02-useast1* instead of *graph* in the United States

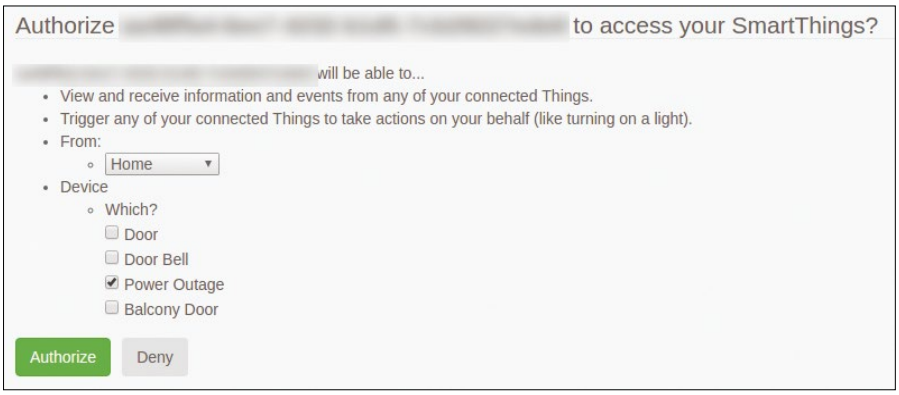


Figure 12: The user can allow web access to selected Z-Wave devices.

and *graph-eu01-euwest1* in the UK; otherwise, the API won't find the user's hub equipment, who is no doubt busy tearing their hair out in frustration. They should be using the industry standard of sharding via the username instead.

However, there is a dedicated community that molds the breadcrumbs uncharitably thrown to it into working applications. The only correct strategy for Samsung has to be opening up the entire project, including the communication protocols used, and leaving it to the community to transform it into a maintainable platform with a future. There is certainly enough potential. ■■■

INFO

[1] "How-to Geek: How to Select a Battery Backup for Your Computer": *http://www.howtogeek.com/161479/how-to-select-a-battery-backup-for-your-computer/*

[2] "IFTTT Home Automation" by Mike Schilli, *Linux Pro Magazine*, issue 189, August 2016, pg. 60, *http://www.linuxpromagazine.com/Issues/2016/189/Perl-IFTTT-Home-Automation*

[3] SmartThings Hub by Samsung: *https://www.smartthings.com*

[4] "Video Preview" by Mike Schilli, *Linux Pro Magazine*, issue 195, February 2017, pg. 52, *http://www.linuxpromagazine.com/Issues/2017/195/Perl-Video-Preview*

[5] SmartThings API tutorial: *http://docs.smartthings.com/en/latest/getting-started/first-smartapp.html*

[6] Listings for this article: *ftp://ftp.linux-magazine.com/pub/listings/magazine/198/*

[7] "WiFi Connect Messages" by Mike Schilli, *Linux Pro Magazine*, issue 186, May 2016, pg. 64, *http://www.linuxpromagazine.com/Issues/2016/186/Perl-WiFi-Connect-Messages*

[8] Prowl: *https://www.prowlapp.com*

```
$ curl -H "Authorization: Bearer xxx" https://graph-na02-useast1.api.smartthings.com/api/smar
tapps/endpoints
[{"oauthClient":{"clientId":"xxx"},"location":{"id":"yyy","name":"Home"},"uri":"https://graph
-na02-useast1.api.smartthings.com:443/api/smartapps/installations/zzz","base_url":"https://gr
aph-na02-useast1.api.smartthings.com:443","url":"/api/smartapps/installations/zzz"}]

$ curl -H "Authorization: Bearer xxx" https://graph-na02-useast1.api.smartthings.com:443/api/
smartapps/installations/zzz/power
{"power":"ok"}
```

Figure 13: Curl at the command line using the web API to access the power outage sensor.

The sys admin's daily grind: Mosquitto

No Insect Bites

Sys admin Charly does not tend toward hostilities, but he has huge problems with mosquitoes. Despite this, he does make an exception for the Mosquitto message broker.

By Charly Kühnast

I recently read that around 100 species become extinct every year. So why not mosquitoes? I am merciless on the issue, because I happen to be allergic to their stings. I can't sleep, with just one of these pests buzzing around my bedroom. My weapon of choice – always at hand to decimate the population of my enemy – is a 30-year-old school atlas beside the bed. But it is still a Sisyphean task.

And the bloodstains on the wallpaper also impact on the kind of atmosphere that I would like have in my bedroom, which explains why I repaper the walls nearly every year. I also failed to secure a majority vote in the family council for my suggestions of adopting mosquito-eating spiders or using a Linux and LiDAR (light detection and ranging) based mosquito laser. War is always miserable.

You will thus understand that my response to a piece of software named after these flying bloodsuckers was somewhat skeptical. However, this much I will reveal in advance, Mosquitto [1] is allowed to stay. The tool was given its name, because the developer wanted to have the letters MQTT in there somewhere, as the acronym for message queue telemetry transport. And telemetry is the right keyword for describing Mosquitto's task.

Mosquitto operates a messaging hub; in MQTT speak, this is known as the message broker. Any client wanting to communicate sends its message to the broker (publisher), and all clients that have a subscription with the broker (subscribers) then receive the message. The broker is simply named `mosquitto` and runs as a daemon; the clients that publish and subscribe go by the names of `mosquitto_pub` and `mosquitto_sub`.

First, I start the broker. It normally runs in the background, but I can also stop the daemon and start it in the foreground on the console. Because Mosquitto runs on port 1883, I don't even need root privileges to do this:

```
mosquitto -v
```

The `-v` parameter sets a higher verbosity level, so that I can send to the broker at work. Then I use the `mosquitto_sub` client to subscribe to a topic. The topic is freely selectable and should ideally describe the type of data that you can expect in a meaningful way. As an example, I can subscribe to the topic `date`:

```
mosquitto_sub -h 127.0.0.1 ➥
  -i subscriber -t date
```

The `-i` parameter describes the client's identity and is freely definable. I used `subscriber` here; in a practical application, however, it would make more sense to use the client's hostname.

Mosquitto News

Time to publish something. On another console, I typed the following:

```
mosquitto_pub -h 127.0.0.1 ➥
  -i publisher -t date -m "30.02.2017 ;-)"
```

See Figure 1. The text passed in with `-m <text>` immediately appears on the console where `mosquitto_sub` is waiting for data.

Impressively simple, don't you think? In daily use, topics are often hierarchically organized; users separate the hierarchy levels with a slash, as in: `readings/humidity/outdoor`. You can secure communication with TLS and also stipulate that the server and clients identify themselves with certificates. I must admit: Mosquitto is useful. But my atlas is still staying where it is! ■■■

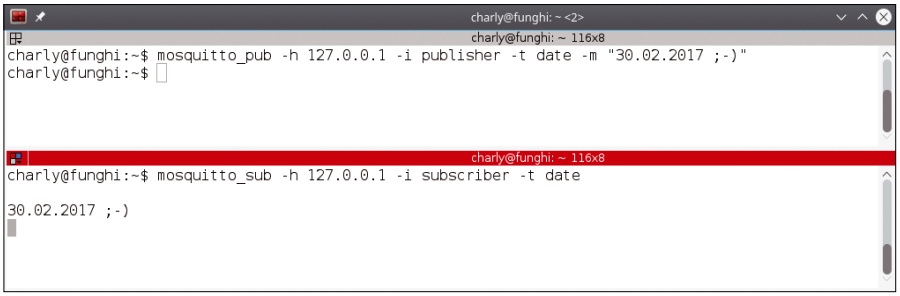


Figure 1: In the upper console, a client is sending a message to another client – the console at the bottom receives it immediately.

INFO

[1] Mosquitto: *https://mosquitto.org*

CHARLY KÜHNAST

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

REAL SOLUTIONS FOR REAL NETWORKS

FREE CD or DVD in Every Issue!

Each issue delivers technical solutions to the real-world problems you face every day.

Learn the latest techniques for better:

- network security
- system management
- troubleshooting
- performance tuning
- virtualization
- cloud computing

on Windows, Linux, Solaris, and popular varieties of Unix.

6 issues per year!

ORDER ONLINE AT: shop.linuxnewmedia.com

Free, secure universal communication software

Ringing In Change

In the last few years, secure text, voice, and video transmission have become major areas of free software development. One of the leaders in this field is Ring. *By Bruce Byfield*

Ring [1] is a GNU communication project started by Savoir-faire Linux [2]. Based in Montreal, Savoir-faire Linux was founded in the late 1990s, when the possibilities of free software were first being widely recognized. "Our bet was that open source software would become a key player of the world's digital economy and that organizations that wouldn't be open source experts would need partners like us," says the Ring team. The bet paid off, and today Savoir-faire employs a multidiscipline team of 150 employees, working with "tools ranging from Red Hat to the Azure cloud" and designing products on every level, from kernel hacking to application and mobile development. In addition to Ring, the company contributes to a wide range of projects, including the Linux kernel, FFmpeg, Debian, Buildroot, and Eclipse.

What makes Ring stand out as a telephony solution is its attention to cutting edge security techniques. Ring as a whole depends on OpenDHT [3], a library developed by Savoir-faire that is similar to the distributed hash table (DHT) used by BitTorrent to find peers sharing a file on a network. Each installation of Ring runs its own DHT node, so users connect peer to peer and control their own accounts and identities; damage to systems from denial of service attacks and natural disaster is therefore contained rather than affecting the entire network.

Ring identifies users via their RSA public key, with control of an account defined by control over a particular private key. Different devices connected to the same account are associated with an x509 certificate chain [4], with each device assigned a new key and certificate pair signed using the main Ring account's private key. The certificate chain is verified each time a device is used to prevent man-in-the-middle attacks.

A somewhat novel aspect of Ring is its use of an Ethereum blockchain [5] – a free software version of the distributed databases used in other distributed systems, such as Bitcoin – as a directory of usernames. According to the Ring team, in the first versions of Ring, users could identify each other only by their public key fingerprints – a 40-character hash that is cumbersome to transmit and difficult for humans to use. By using a blockchain, in newer versions of Ring, users can register unique usernames without the need for a centralized database.

Information is shared through Ring with perfect forward secrecy (PFS) protocols [6], which change the exchanged keys in each session. Media streams do much the same, using a Secure Real-Time Transport Protocol (SRTP) [7].

The User Experience

Such details have been a major obstacle to the general use of security and encryption. As the Ring team notes, users are accustomed to weak security, such as passwords on email, and can be reluctant to change their habits, even when the necessity is obvious. In fact, a truism in security circles is that, when asked to choose between convenience and security, users will almost always choose convenience. In this situation, the challenge for the developers of applications like Ring is to provide the features that users expect without compromising the distributed network and encryption.

Ring is currently in beta release, but it is already well on its way to balancing these requirements. Ring can be downloaded for Android, GNU/Linux, Mac OS X, and Windows, along with

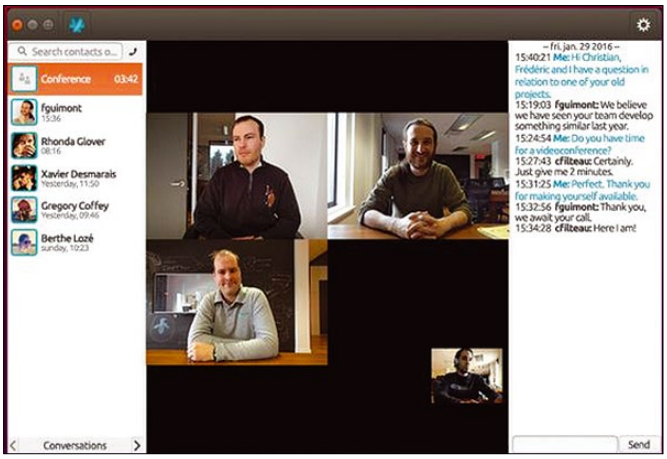


Figure 1: Ring promises to be a leader in secure messaging technology.

full instructions for each [8]. For GNU/Linux, recent releases of Debian, Ubuntu, and Fedora are officially supported (Figure 1), with options for installation from the project site, packages, and tarballs. The only potential problem is that releases are not upgradable, which means that an installed version must be removed before an upgrade is installed. If not, the users of one version cannot communicate with the users of another.

Once installed, users must register with the blockchain, using the starting wizard to register an avatar, user name, and password (Figure 2). Users may also want to adjust the settings, configuring such features as notifications for calls and chat, and the position of the chat window on the desktop. Users may also want to install Ring on another device, associating it with the already installed account and adding a PIN, so they can receive incoming calls without being on a particular device (Figure 3).

With this setup, Ring works much like Signal or the standard Android apps for voice and chat messaging. However, you might want to study Ring's other features [9] before you need them. The feature list includes putting a call on hold, disabling the microphone and video camera, recording audio, and sharing screens and files.

Overall, Ring is not as easy to use as the apps on the average phone. However, the beta hides much of the complexity from average users and is a promising indicator of what the general release will be like. The biggest challenge, says the Ring team, is to provide the same features across different platforms.

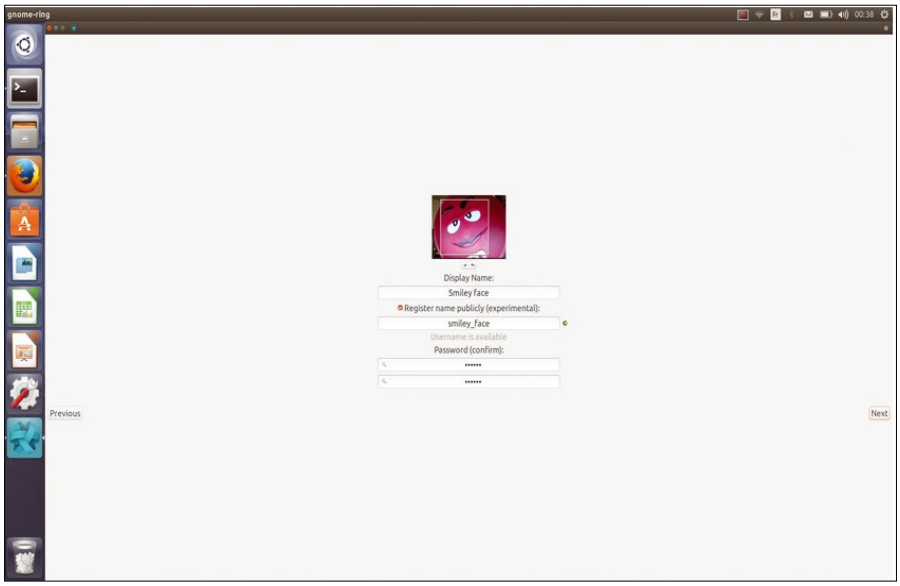


Figure 2: Ring uses blockchain as a directory of usernames.

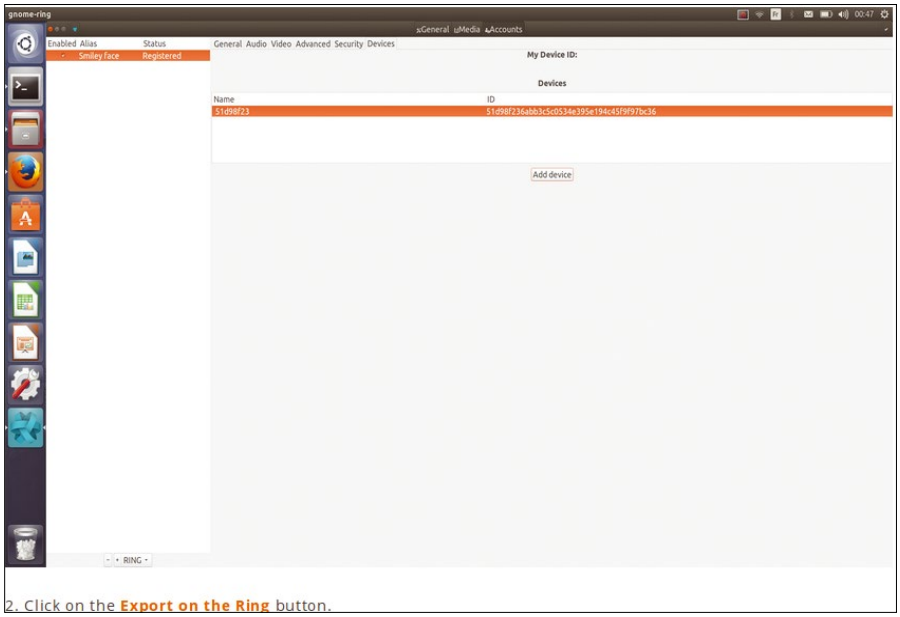


Figure 3: Ring can associate multiple devices with the same account.

Future Development

Ring already includes many features that similar software does not. However, before general release, Savoir-faire hopes to add such features as content management; synchronized contact lists; history, group chat, and enhanced file sharing; and improved encryption for texting. "In general," says the Ring team, "we work to make Ring as user friendly as possible while preserving user privacy and the distributed nature of Ring."

Future plans include bringing Ring to embedded systems, smart TVs, and other Internet of Things devices – all supported by documentation.

Ring is an ambitious effort that appears to be making a successful transition from a walled garden to a public project. Although it has not reached the level of simplicity for the end user as its rival Signal, it is not far off and offers more features, as well. If Ring can develop its user interface as well as it has its behind-the-scenes technology, it has a strong chance of becoming one of the leading free software tools of the next few years. ■■■

INFO

[1] Ring: *https://ring.cx/*

[2] Savoir-faire Linux: *https://www.savoirfairelinux.com/en/home*

[3] OpenDHT: *https://blog.savoirfairelinux.com/en-ca/s/opendht/*

[4] x509 certificate chain: *https://en.wikipedia.org/wiki/X.509#Certificate_chains_and_cross-certification*

[5] Ethereum blockchain: *https://en.wikipedia.org/wiki/Ethereum*

[6] PFS: *https://en.wikipedia.org/wiki/Forward_secrecy*

[7] SRTP: *https://en.wikipedia.org/wiki/Secure_Real-time_Transport_Protocol*

[8] Downloads: *https://ring.cx/en/download*

[9] Useful functions: *https://ring.cx/en/documentation/useful-functions-for-a-call-in-progress-on-gnulinux*

Ben Everard

Recently, I had to deal with a server that had a corrupted `sudoers` file. It's a bit problematic because you need `sudo` permissions to correct the problem, but you can't get `sudo` permissions until you fix it. After a bit of nervous to-ing and fro-ing, I came across the `pkexec` command, which solves the problem by using polkit to grant permissions. Phew. It reminded me of another time I was dealing with the aftermath of a security breach and the whole of `/usr/bin` was corrupted. We had to find alternatives for all the commands until we had everything fixed (for example, rather than using `ls`, you can type `./` then tap tab twice). There are a lot of ways of doing things in Linux, and this can make it a little complex sometimes, but it also means that when one way goes wrong, you have other options. These alternative approaches can end up helping in ways that are almost impossible to foresee. I thought about this plurality of options as I read Mike's article on browsing the web via the command line. Yes, it seems a little eccentric when you first think about it, but it can come in useful in many ways, not least when you mess up your graphics drivers.

In this month's Core Technology, Valentine Sinitsyn also takes a look at some more alternatives, specifically with regard to how authentication works (although as responsible admins, we should probably consider `sudo` or `su` to be the alternative rather than the default). Simon Phipps is also looking at alternatives, particularly with the nomenclature around Libre, Free, and Open Source.

The range of different options available to users of Linux is a huge advantage. It doesn't really matter what words we use when advocating for Free Software, fixing an unusual situation, or just trying to browse the web while having a drink at an airport. So turn the page, dive in, and join us as we celebrate the diversity of technology in the Linux ecosystem.

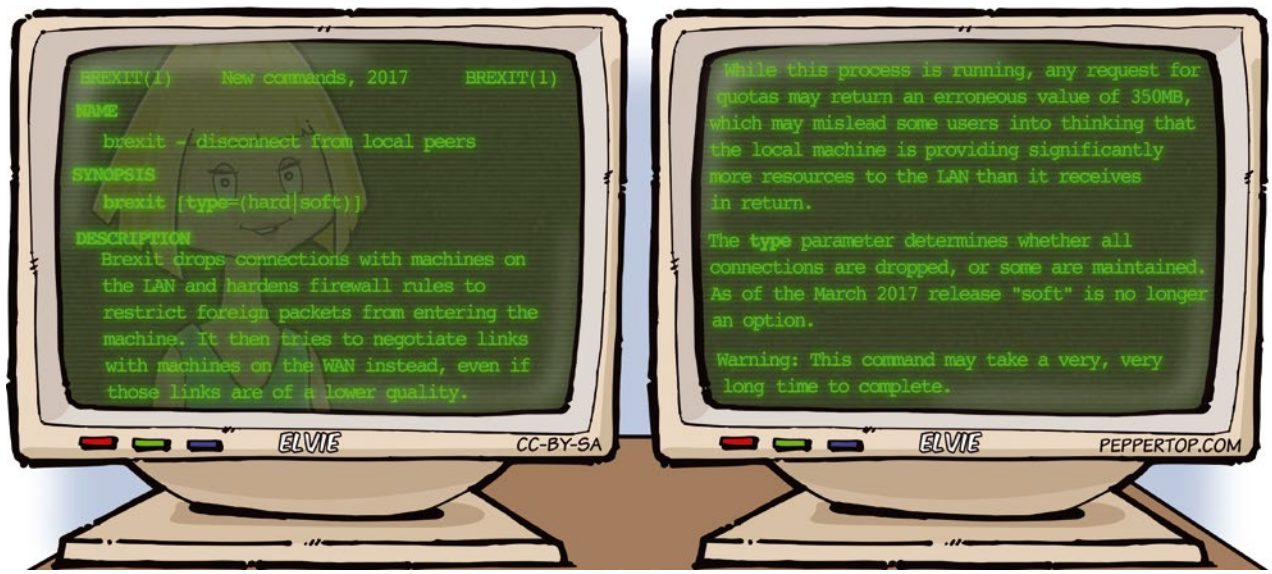
– Ben Everard

Andrew Gregory

Graham Morrison

Mike Saunders

LINUXVOICE



NEWSANALYSIS

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

Opinion

The Shallowness of the Free vs. Open Dichotomy

No matter what the zealots say, Free Software and Open Source occupy common ground. BY SIMON PHIPPS

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

The term *Open Source*, in the context of software, was coined in 1998 by a group of experienced software freedom advocates frustrated by the challenges of helping corporations adopt Free Software. As the movement has energetically grown over the ensuing decades, it has been repeatedly necessary to remind people that framing it as a *methodology* is a construct chosen nearly 20 years ago to help cultivate executive acceptance and business promotion of software freedom. The frame is necessarily not the entire story, no matter how often newly-woke geeks may assert it should be and how evil it is not to say "Free Software." Open Source is inescapably a part of the culture, philosophy, and ethical construct that is software freedom, not an alternative to it.

Here's why. Corporations are not people, and so can't "behave ethically" – doing so requires consciousness as a minimum. The people they employ can be expected to behave ethically, but a corporation will follow its programming to optimize the objectives stated in its bylaws. The people tending the machine can steer it towards different ways of achieving those objectives and can express their ethical selves through their choices, but they are not free to justify preferences purely on the basis of ethics. As a consequence, most advocacy of Open Source has focused on helping those corporate employees demonstrate the value arising from it rather than the values motivating the people involved with it.

> The people tending the machine can steer it towards different ways of achieving those objectives and can express their ethical selves through their choices, but they are not free to justify preferences purely on the basis of ethics.

This pragmatism has been ceaselessly criticized by people adhering to the supposed "purity" of the term "Free Software," who attempt to claim that Open Source and Free Software are different things and the advocates of Open Source are at best amoral. They are not; effective adoption of Open Source involves the principled application of Free Software. More than that, there's a strong causal relationship between software freedom and the value business deployers gain from Open Source software.

To seek the benefits without embracing the values is possible but inadvisable. Prices can be cut artificially as an incentive; documentation, architectures, APIs, and even code snapshots can be delivered on demand by proprietary vendors. But if you are not the one enjoying software freedom, all those benefits are contingent on your relationship with the one who is. Those values also have wider applicability. They protect against covert abuses and also lead one to shun the infringement of liberties.

None of this is peculiar to Open Source. It is equally possible for the term "Free Software" to be used for its benefits without embracing its values, and it's actually easier for the newcomer to interpret the term at face value and assume price is the primary motivation. Indeed, that accidental invocation of the "price frame" continues to lead people astray even today. So renewed moves to define Open Source and Free Software as somehow different are mistaken. What's needed is to reconnect users of Open Source Free Software with the origin of the benefits they enjoy from it. That origin is software freedom, the certainty of being explicitly entitled to use, improve, and share the software upon which you depend. ■■■

Show Me the Money

How do you charge for something that is free? Just ask!

BY ANDREW GREGORY

As the founder of a modestly successful crowdfunded venture, I applaud the team behind AppCenter, the pay-what-you-want app store for ElementaryOS that, at the time of this writing, has raised $9,570, comfortably meeting its funding target of $8,000.

The market system of incentives is fundamentally changed by Free Software, as the consumer doesn't pay, regardless of how much he or she values the work of the developers. If Canonical, for example, were to start charging for Ubuntu, any number of substitute distros could be installed free of charge in its place. It doesn't matter that Ubuntu is key to the financial success of such Internet behemoths as Amazon Web Services; the marginal cost of a replacement is £0, so the relationship between price and worth is broken.

Canonical can fight back by using Ubuntu to create a market for a new product: its support services. But Canonical can only do this because it's big enough to do so. Small distro developers can't alter the market, so if they want money, they need to ask for it.

That's exactly why so many previous attempts at app stores for Linux have failed. Nobody's going to pay for Free Software on the desktop in a purely capitalistic transaction (buy me a beer, and I'll tell you why classical economic theory is flawed because there's no such thing as "the free market"). But take out the transactional element and turn it into goodwill, into membership of a club, into feeling like you're doing the right thing; then it's just possible that the little guys have the edge over big players like Canonical.

ElementaryOS, for all its polish (and it is an extremely well-put-together desktop OS), is a small project. Its developers deserve recognition. The Linux community as a whole should be rooting for it, because the more it flourishes, the more consumers will look up from their shiny Apple devices and be tempted to try something new.

Lots of developers give you the chance to contribute something for their work. Whether it's on the AppCenter, on Patreon, or on one-off buttons called something like "Buy the creator a coffee." Good luck to them: If they can make a few quid without pressuring me with a hard sell, I'll be more inclined to give them my support than if they were making it impossible to use the software without handing over my credit card details. It's tricky, but the right balance is out there for those brave enough to look for it. ■■■

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

MADDOG'S DOGHOUSE

Free hardware is a noble concept, but expenses associated with the hardware manufacturing process means your single-board system will never be quite as a free as the software that runs on it. BY JON "MADDOG" HALL

FOSS and FOH

From time to time, I hear people talk about "Free" or "Open" Hardware and what it means to them. After all, we have Free and Open Source Software (FOSS) and shouldn't some of the same ideas of collaboration that surround FOSS apply to hardware? In many ways, Free and Open Hardware (FOH) could be like FOSS. Creators of hardware could collaborate with potential users to help determine the characteristics of the hardware. Developers who want to use the hardware to make a product could help test the functionality and interfaces of the hardware to make sure it actually works and meets their needs.

However some of the features of FOSS do not apply equally to hardware. For example, if you give away a copy of your hardware, it is a physical thing you are giving away, and, unlike software, it probably will cost money to replace it.

Costs to prototype and manufacture hardware have dropped over the years (as has the cost of the hardware itself), so it is easier than ever to design hardware outside of a large company. However, there are still significant costs for the development of hardware. Let's look at some of the costs of developing a single board computer (SBC). I will warn you that this will be from a very high-level view.

First, you should decide what features your SBC will have. What architecture should the CPU have, what Graphical Processing Unit (GPU) will it have, how much RAM memory, what I/O peripherals, power supply, and so forth. You might generate a list of features from looking at other SBCs in the marketplace.

Then, if you have the knowledge and skill, you either find or create a circuit diagram that implements the features you desire. Sometimes you might find a very expensive board with lots of features and you decide that you can save a lot of money, heat, power usage, etc. by "depopulating" an existing design to fit your needs.

Eventually, you will have a design that you like, so you may try to create a printed circuit board with your parts on it. This is where the design may start to get expensive. Although many of the parts (particularly the resisters, capacitors, and other "passive components") are very inexpensive, the "active components" (RAM, CPU/GPU, etc.) can be fairly expensive and can also be ruined by static, heat, and other environmental issues. Many components are also "surface mount," meaning that they only sit on pads of connectivity that are very, very small and require a steady hand for placement OR a machine called a Surface Mount Technology (SMT) machine (or its smaller cousin, the pick and place machine). A world class SMT machine can cost well over four million dollars and place 60,000 components per hour. To set up an SMT may cost between $100,000 and $400,000. A "pick and place" machine that sits on a table top could cost several thousand dollars and place fewer components slower, but is more practical for smaller numbers of boards or prototypes.

Another issue is getting access to the documentation for the active components. Often this documentation is not available to the public, and you have to pay the manufacturer to get this information, or perhaps you can get access to the information through a distributor who has a partnership with the vendor.

Perhaps you now have a working board. You test it and it seems to do everything you want. Does it have WiFi? If so, you might need to get the wireless certification from your national certification agency, and this might cost tens of thousands of dollars. Depending on the target market, you might have many more certifications to test, patent royalties to pay, and other regulatory costs.

Also, you might need an operating system, or a specific distribution of an OS, to be ported to your board. In order to port the board (assuming you do not also have those skills), you might have to generate several of your boards to lend to software people to finish the port. Now you have a finished board with some distributions ported to it. Assuming you want to build a lot of the boards to use in products, you have to raise the money to purchase the parts for large-scale manufacturing, as the parts suppliers will want to be paid before they release the parts for manufacture.

After that, you only have to purchase the packaging, decide how to ship the finished product to customers, and determine how to take back boards damaged in manufacturing and shipping.

Development costs have come down, but a new SBC can still cost between three to four million dollars, taking into account everything necessary to release the product to the customer. As you can see, while collaboration is fine, free hardware is not as free as free software. ■■■

Back to Basics: Discover the World of IRC

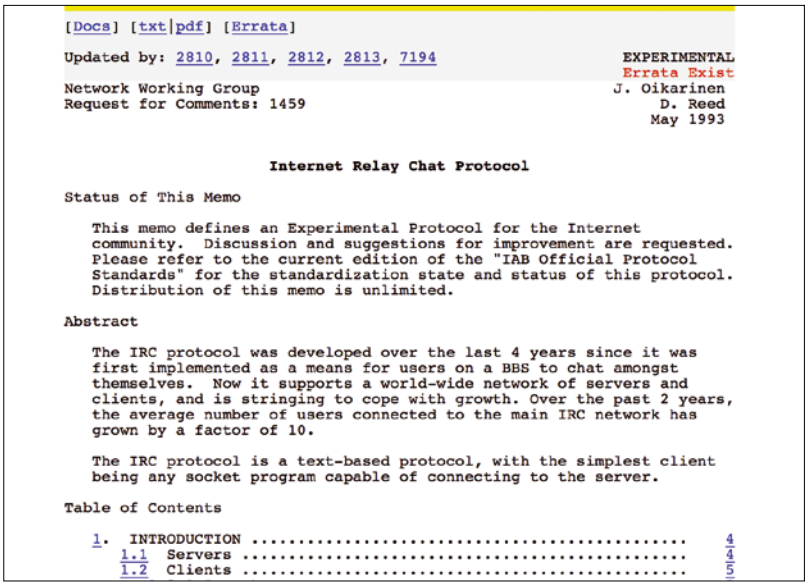
Drop Discord. Say goodbye to Slack. The real way to communicate online is IRC – here's why it still rocks.

BY MIKE SAUNDERS

The number of chat and discussion services competing for our attention is growing at a bewildering pace. Take a typical smartphone and its methods for communication: It'll probably have SMS, Facebook Messenger, Google Hangouts, WhatsApp, and Telegram. Then, add a few phone-specific services on top. Every other day a new app arrives promising to be the ultimate solution for all of our communication woes – but it just becomes another annoyance filling the notification bar.

And there's another problem: Most of these services are proprietary and centralized. If WhatsApp's servers go down (unlikely, as they're running on FreeBSD), then there's nothing that the service's vast user base can do. Well, apart from enjoying a few moments of peace and quiet, of course. The Internet was created as a distributed network that could route around problems, yet we're increasingly reliant on highly centralized services run by single companies. It's not how things were meant to be.

So, in this article, I'll examine the state of Internet Relay Chat (IRC). This is an open protocol for real-time, text-based communication that is older than the web, and most importantly, it's decentralized. Anyone can run IRC server software, and it's possible to add multiple computers to the same IRC server to create a network, providing backups in multiple locations in case one machine happens to go down. IRC may not offer all the fancy bells and whistles of other communication services, but it's proven, reliable, and still widely used – especially in the free and open source software community.



[Docs] [txt|pdf] [Errata]

Updated by: 2810, 2811, 2812, 2813, 7194

EXPERIMENTAL
Errata Exist

Network Working Group
Request for Comments: 1459

J. Oikarinen
D. Reed
May 1993

Internet Relay Chat Protocol

Status of This Memo

This memo defines an Experimental Protocol for the Internet community. Discussion and suggestions for improvement are requested. Please refer to the current edition of the "IAB Official Protocol Standards" for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Abstract

The IRC protocol was developed over the last 4 years since it was first implemented as a means for users on a BBS to chat amongst themselves. Now it supports a world-wide network of servers and clients, and is stringing to cope with growth. Over the past 2 years, the average number of users connected to the main IRC network has grown by a factor of 10.

The IRC protocol is a text-based protocol, with the simplest client being any socket program capable of connecting to the server.

Table of Contents

1. INTRODUCTION 4
1.1 Servers 4
1.2 Clients 5

Figure 1: The IRC protocol is open and defined in an RFC from 1993.

A Bit of Background

IRC was originally developed in Finland in 1988, as an attempt to expand a bulletin board system (BBS) with real-time chat features. One year later, 40 IRC servers across the world were in operation; that might not sound like a huge figure, but bear in mind that this was before the web, and very few people had Internet access at home. Multiple distinct IRC networks popped up, offering their own sets of features and topics, and it soon looked like IRC could become fragmented.

Fortunately, a standard for the protocol was created in May 1993 as RFC 1459 [1]. An RFC is a Request For Comments – a document by the Internet Engineering Task Force (IETF) that aims to provide a reference for future implementations of a protocol. So, despite a few early bumps in the road, IRC got off to a good start and became the most popular way to communicate online in the late 1990s (Figure 1).

Some of the most popular IRC networks of the time were Dalnet, EFnet, and IRCnet. Today, one of the most popular networks in the FOSS world is Freenode, which is hosted by 32 servers working in tandem. When you connect to Freenode, you connect to one of those 32 servers on the network, but as they're linked together you can still communicate in real-time with users connected to a different machine on Freenode. At any given time, there are about 90,000 users on Freenode, chatting in about 40,000 channels.

But what exactly is a channel? Think of it as a chat room – a place with a specific topic where a bunch of people get together to discuss it. For instance, Freenode has an Ubuntu channel to ask

questions and discuss that distro; there are channels for LibreOffice, the Linux kernel, and many other FOSS-related topics as well. Anyone can create a new channel and invite users to it, although the vast majority of the 40,000 channels on Freenode are silent or virtually empty.

IRC is popular in geek circles for many reasons. Although it's real-time – so when you type a message everyone else in the channel sees it pretty much straight away – it's not like a team call where you're expected to answer at any moment. IRC users have nicknames and can direct messages at each other using these. So, you could be in a very busy IRC channel and not pay much attention to the general chat until someone mentions your nickname specifically, in which case your IRC software (aka "client") will notify you in some way. This could be a sound, or bold text on the specific line, or something in your notification area.

Take the LibreOffice team as an example: There are channels for development, QA, design, documentation, and other topics. Many LibreOffice contributors join all of these channels, but rarely take part in conversations unless they are specifically asked a question. It's also common for participants to add "_afk" to their nicknames – which means "away for now" – so other users know not to expect an immediate response.

Going Hands-On

That's enough background; let's see IRC in action. Because IRC is an open protocol, anyone can implement a client program to access a network. And – as you'd expect in the FOSS world – there's a huge range of clients to choose from. Our favorite graphical client is HexChat [2] (Figure 2), but we recommend learning the text-based commands for using IRC, so install the very awesome Irssi [3] client in your distro's package manager and start it up in a terminal by entering `irssi`.

You'll see… not a lot. By default, Irssi will sit there waiting for you to issue commands, and the first thing you need to do is connect to an IRC server. So enter this:

```
/server chat.freenode.net
```

Here you can see that IRC commands begin with forward slashes, followed by parameters. All we do here is tell Irssi to establish a connection to the IRC server (daemon) running on the host chat.freenode.net. Irssi does a bit of handshaking with the server, identifies itself, and then we're connected.

But still, not much else is going on. We can't chat to anyone at the moment, because we haven't joined any channels. Let's fix that:

```
/join #ubuntu
```

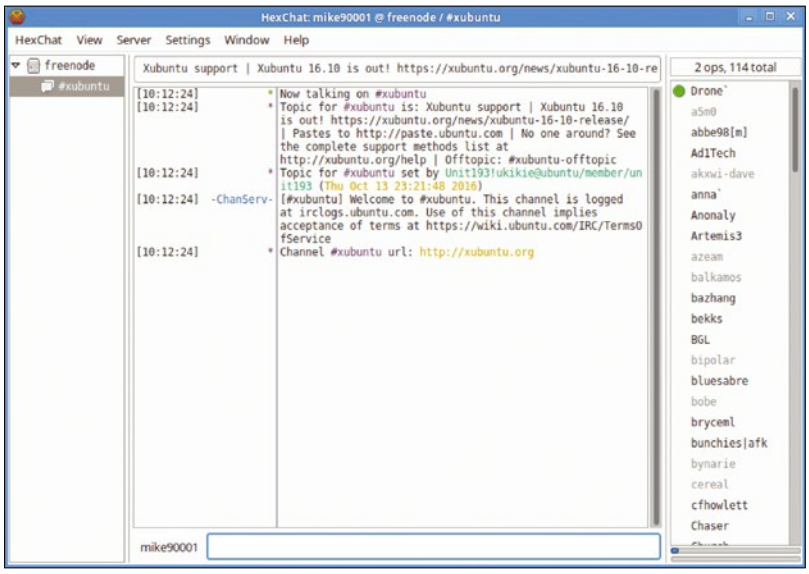


Figure 2: If you prefer a graphical IRC client, HexChat (based on XChat) is one we highly recommend.

Note that channels are typically prefixed with hash (#) marks. This command opens up a new window in Irssi; you'll see the topic of the channel at the top, providing some information about the latest release and links to useful information (Figure 3). It's good IRC etiquette to always read the topic of a channel when you join – especially if something important has just occurred and everyone is talking about it. (See the "IRC Etiquette" box for more information.)

Next, you'll see a list of nicknames, surrounded by square brackets. These are participants in the channel, and those with @ characters before their names are operators (aka ops) – they have special privileges to change the topic, ban users, and perform other jobs.

The next thing you'll want to do is set your nickname. Use the following:

```
/nick SomeNameHere
```

Use whatever name you like, but if it's already taken, you'll have to choose something else! (To register

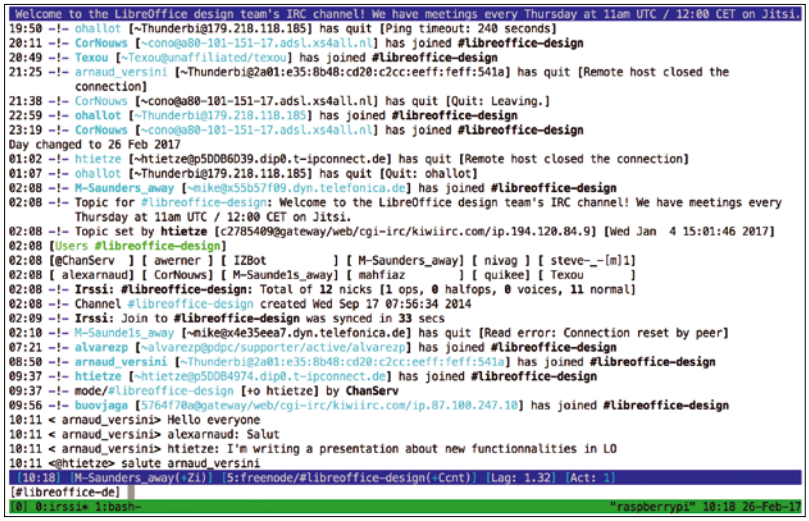


Figure 3: Irssi may look a bit cluttered when you first start using it, but you'll soon come to grips with its interface.

IRC Etiquette

In general, good behavior on IRC is the same as elsewhere on the Internet: don't troll, don't type in ALL CAPS, and be patient if you ask a question. One big no-no on IRC is "flooding" – that is, pasting lots of text into your IRC client. You might be tempted to do this if you have something genuinely useful to contribute, such as some code or a chunk of a log file, but it's very bad practice. Users who are temporarily away from the channel and switch back to check what's going on will just see your pasted text and have to scroll around to see what the context is. Instead, use a site like Pastebin [4] and then send the link to the IRC channel.

When you're joining a new channel, it's a good idea to watch the channel for a few minutes to get a feel for the discussion and general behavior of the participants. You can say hello and introduce yourself on smaller channels; on larger ones, like #ubuntu, that could be interpreted by some users as overly trivial and distracting from real conversation. If you want to ask a question but aren't sure whether it's suitable for a particular channel, you can ask politely beforehand: "Hi – are questions about X allowed here?"

your nickname so that only you can use it, see the "Registering a Nickname" box.) Now you can start chatting away – any text you type, followed by Enter, will appear in the channel for all to see. If you want to alert a specific user in the channel, you can type the first few letters of his/her nickname and use tab completion. Then, you can type something like the following to notify them:

```
SomeGuy: Have you downloaded the latest release?
```

You can also join other channels – there's a good list of the most popular ones (and a search tool) on the web [5] (Figure 4). Try searching for topics that interest you – software, programming languages, music, sports, and more. You'll see the channel names along with the number of participants, so you can immediately get an impression of how busy the channels are.

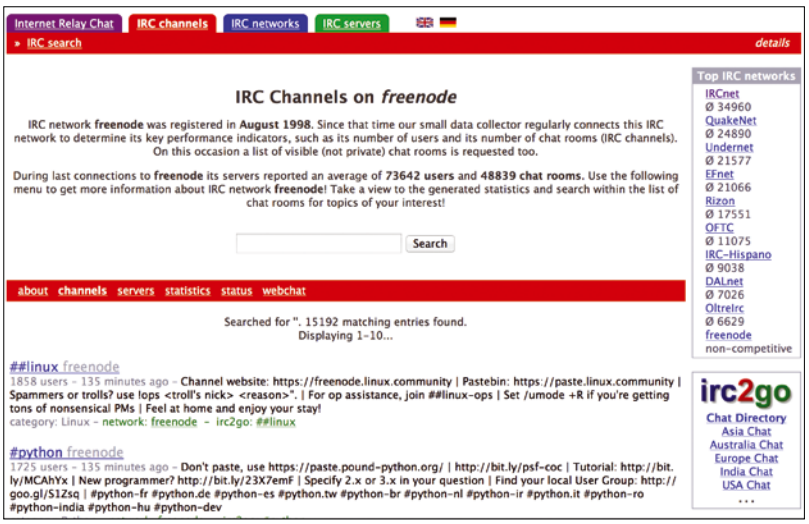


Figure 4: On irc.netsplit.de, you can search for channels covering various topics across many IRC networks.

Meanwhile, you can use Alt+P and Alt+N to scroll back and forward through the chats – this is very helpful if there's a lot of activity going on and you need to catch up. In the Irssi status bar, you'll see things like this:

```
[Act: 2]
```

That means there is activity in the second window. If the number is purple, someone has mentioned your nickname – so you'll probably want to switch to that window, read the message, and respond accordingly. To close a window, enter `/wc`, and use `/quit` to disconnect from the IRC network and terminate Irssi.

Super Scripts

Irssi is a great IRC client on its own, but it can be expanded with extra features and functionality using Perl scripts (Figure 5). These should be placed inside your `.irssi/scripts/` in your home directory and can be loaded using the `/script load` command followed by the script name (omitting the `.pl` extension). Before we try some scripts, let's see how they work. In `~/.irssi/scripts/`, create a text file called `foo.pl` with the following contents:

```
use Irssi;

Irssi::command_bind hello => sub {
    print 'Hello, world!';
};
```

This is a very simple script that adds an extra command to Irssi: `/hello`. As you can see, it simply prints the text "Hello, world" to the screen (note that it doesn't actually send it to a chat channel). With this file saved, go back to Irssi and then enter the following commands:

```
/script load foo
/hello
```

Et voilà – the "Hello, world" message appears on your screen. If you're already a dab hand at Perl,

you can try to write some more complex scripts; there's a useful introduction tutorial online [6].

Now, on the official Irssi scripts site [7], you'll find a huge number of scripts to do very cool (and sometimes pointless) things in IRC. By default, these are listed alphabetically, but if you click the Votes link on the right-hand side a couple of times, it will show the scripts with the most votes at the top. One of our favorite scripts is the mightily useful Trigger [8], which is like a Swiss army knife of scripts. It lets you define commands to be executed when certain events or actions happen in IRC – effectively letting you create your own automated IRC bot without having to learn lots of scripting.

When you have it installed and loaded, enter `/trigger help` to get an overview of what it can do. You can use regular expressions to match certain conditions and then perform an action automatically, such as banning a user if he or she uses certain words (e.g., swearing).

Another useful script is `adv_windowlist`, which as the name suggests provides an advanced window list inside the IRC client. When you load this, after a few moments a list of available windows is displayed down the left-hand side of your screen; they are also colored accordingly depending on activity. If you have the screen space, it's well worth installing, as it makes it much easier to navigate between windows (rather than trying to keep track of things in the rather cluttered status bar).

Then there's trackbar22. This is truly a godsend if you monitor multiple channels, as it shows you the exact point where you last looked at a window. For example, say you're watching the #ubuntu channel but then you switch to a different window for a few minutes. When you return to #ubuntu, if there has been a lot of activity in the meantime, you'd have to scroll around a while and try to remember where you last saw the discussion. This isn't always easy, but with trackbar22, you'll see a horizontal line showing exactly when you stopped viewing that window. So, simply scroll up to the line, and you can catch up at the exact point where you left the discussion.

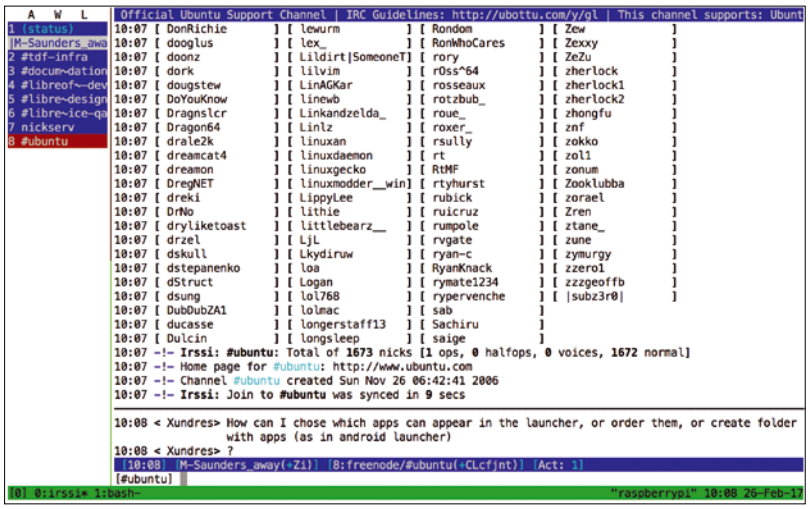


Figure 5: Here's Irssi with some extra scripts installed to provide an advanced window list and bar showing recent activity.

There are many other scripts worth investigating, some of which beef up Irssi with advanced user interface features and make it easier to keep track of busy discussions. Enjoy your time on IRC, and let us know if you discover any especially awesome Linux and FOSS-related channels that we should mention in the magazine! ■■■

Info

[1] RFC 1459: *https://tools.ietf.org/html/rfc1459*

[2] HexChat: *https://hexchat.github.io*

[3] Irssi: *https://irssi.org*

[4] Pastebin: *http://pastebin.com*

[5] Search tool – irc.netsplit.de: *http://irc.netsplit.de/channels/?net=freenode*

[6] Tutorial: *https://juerd.nl/site.plp/irssiscripttut*

[7] Irssi scripts: *https://scripts.irssi.org*

[8] Trigger: *http://wouter.coekaerts.be/irssi/trigger*

Registering a Nickname

Most IRC networks let you register a nickname so that only you can use it. This is important to prevent other users from pretending to be you, causing trouble and trolling. To register, you need to communicate with a special bot user called "nickserv" – the nickname server. This isn't a real person but simply handles the job of registering and confirming nicknames.

Once you've got a nickname you like, enter this:

```
/msg nickserv register <password> <email address>
```

This sends the "register" command to the nickname server, with a password that you specify and your email address. A validation message will be sent to your inbox, so check it and then enter the verification command provided.

From here on, whenever you log on to the IRC network you can identify yourself like so:

```
/msg nickserv identify <nickname> <password>
```

If anyone else tries to use your registered nickname when you're not online, nickserv will tell them that they need to identify first – so you're protected.

FAQ LEDE

Ben Everard investigates the future of Linux-based routers.

BY BEN EVERARD

Q Unless this is about the municipality in the East Flanders region of Belgium, I don't know what we're talking about here.

A Nope. This is the Linux Embedded Development Environment (LEDE), which describes itself as a Linux operating system based on OpenWrt. LEDE [1] is a complete replacement for the vendor-supplied firmware of a wide range of wireless routers and non-network devices.

Q Hang on, isn't OpenWrt itself a complete replacement for the vendor-supplied firmware of a wide range of wireless routers and non-network devices?

A Erm, yes.

Q So, what exactly does LEDE bring to the table?

A The difference isn't so much technical as organizational at the moment. The LEDE project was launched publicly on May 3, 2016 with an announcement on the OpenWrt mailing list where they describe it as a "spin-off" from OpenWrt. They're developing off the OpenWrt codebase but building a new project.

Q It sounds a bit like a fork to me.

A It does seem that way.

Q So, why exactly did these developers spin off from OpenWrt?

A They gave five issues that they were unable to fix from within the OpenWrt community. These (quoted from the announcement) are:

1. Number of active core developers at an all time low, no process for getting more new people involved.
2. Unreliable infrastructure, fixes prevented by internal disagreements and single points of failure.
3. Lack of communication, transparency, and coordination in the OpenWrt project, both inside the core team and between the core team and the rest of the community.
4. Not enough people with commit access to handle the incoming flow of patches, too little attention to testing and regular builds.
5. Lack of focus on stability and documentation.

Q Those all seem like important things to fix, but were they really problems?

A Well, to quote now from the official response from OpenWrt: "While we recognize the current OpenWrt project suffers from a number of issues outlined by Jo-Philip, in each of the 5 bullet points, we do not agree with the conclusions withdrawn, and even less so in deciding to spin off the OpenWrt project in the first place as a way to fix the project and its community."

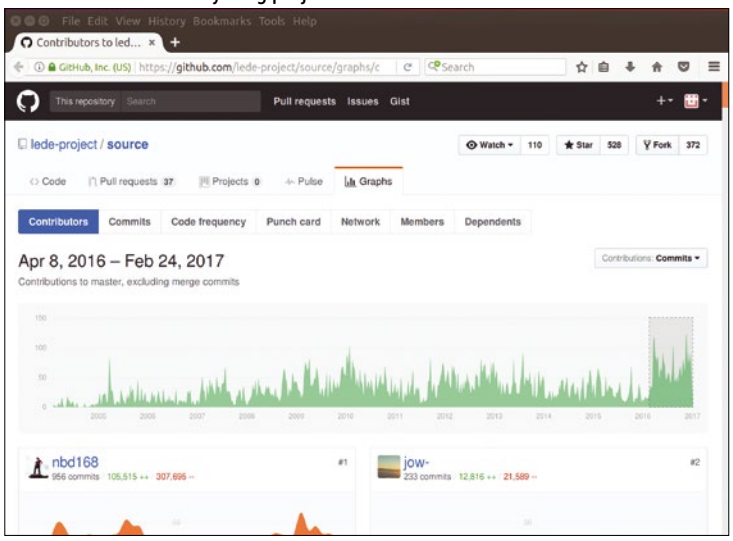
So, yes, those are real problems. It's always hard to know when, if ever, problems are severe enough to mean forking… ahem, I mean spinning off, is necessary. Ultimately, in the open source world, every developer has the right to choose which projects to spend their own time working on, and if that's a fork, sorry, I mean, spin-off, then that's their choice.

Q The LEDE project announcement talked quite a bit about community and contributions. How has that gone since the split?

A Well, thanks to the magic of the Git source code management system, we can see in detail what code has changed in both projects going all the way back to 2005. It's obvious from even a quick glance that in the six months following the fork (sorry, spin-off), the LEDE project was busier than OpenWrt ever had been (Figure 1). In the OpenWrt project, barring a spike in June 2016, the same period was the quietest it has ever been. In that time period, 38 people made 10 or more changes to the LEDE project, whereas only 11 people made that many changes to OpenWrt. Comparing this with same length of time before the split, only 9 people made 10 or more changes to the pre-forked (sorry, spun-off) project.

There are some caveats in that. For example, many of those changes will have been things like changing the project name and website in source code files. However, these are quick changes that have probably been automated. What we see if we look at the commits is sustained activity over

Figure 1: The LEDE GitHub project contribution graphs show that there is momentum behind the young project.



the time period, and this points to new vigor in the development community.

Q So, should we expect LEDE to rapidly outpace OpenWrt as the open source router OS of choice?

A Probably. Extrapolation is always risky, and it's possible that what we're seeing with LEDE is nothing more than the honeymoon phase of a project that will quickly die back; however, we suspect that what we're seeing is the start of a more active community.

Q If it's obvious that the LEDE project's approach to community management is working, won't the rest of the OpenWrt project merge back with them to reform as a single, but more active, community?

A That's certainly possible. There have been peace talks between the two projects, and in December 2016 they published the minutes of the discussions on the OpenWrt mailing list. It looked like the two sides were close to an agreement on merging; you can read the full details online [2]. However, since then, the talks of reconciliation appear to have fallen silent. In February 2017, LEDE published the first version of their OS (17.01). An act that some might see as hostile to the potential peace between the projects.

Q Hang on. Aren't you overdramatizing this a bit? Is publishing a version of software really 'hostile to the potential peace'?

A As Thomas Mann said: "Everything is politics." Pushing a release out is getting the LEDE name known in the wider community and further empowering the fledgling community.

Q I think you're straying a bit too far into gossip magazine territory there. What about the technology?

A Ah yes. Before we got sidetracked by the soap opera drama that is project mailing lists, we mentioned that LEDE is "a complete replacement for the vendor-supplied firmware of a wide range of wireless routers and non-network devices." In other words, many routers for home Internet run operating systems based on Linux, and in some cases it's possible to change the software they're running to a different OS. LEDE (and OpenWrt) are Linux distributions designed to be run on home routers in place of the software they come with.

Q I've always found that my home routers "just work." Why would I want to change them for anything else?

A While home routers can be reliable and easy to use, they often lack a lot of features. For example, with LEDE or OpenWrt you can change the DNS and other network settings, share files directly from the router or even block web adverts. Basically, your router will become a little Linux computer that you can do what you want with, whether that's controlling your network, or hosting some network service.

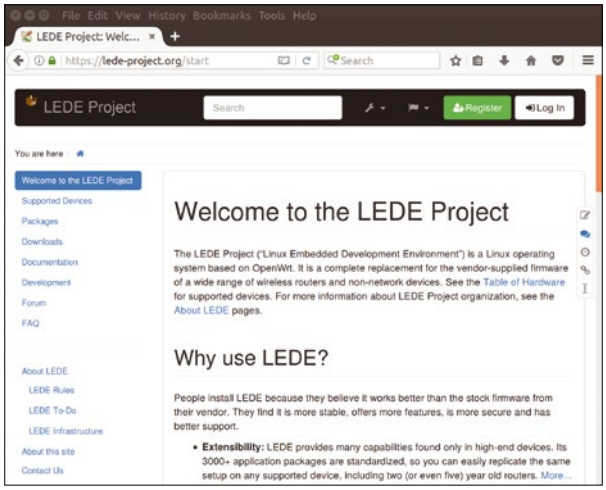
Q Ah, I'd never even thought of doing that on my router. What hardware does it work on?

A There's a list of hardware known to work on LEDE on the project website [3]. If you're the sort of person who likes to do things from scratch, you can start with a Raspberry Pi and build up from there (Figure 2).

Q Right, I'm off to the shops to buy myself a new router. See you next month.

A Bye. ■■■

Figure 2: Find out how to build your own open source router with LEDE.



Info

[1] LEDE project: *https://lede-project.org/*

[2] Mailing list discussion: *http://lists.infradead.org/pipermail/lede-dev/2016-December/004786.html*

[3] Table of hardware: *https://lede-project.org/toh/start*

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

CORE TECHNOLOGY

Everyone wants to be root, because root can do anything. But in fact, its powers are now split. Learn more in this overview of capability sets. BY VALENTINE SINITSYN

Capabilities

Today's Linux is somewhat like a famous sightseeing city you might have visited on your last vacation. There is a historic part that's of no practical use now, yet it is what keeps the city's identity. There are some well-known tourist spots that everybody seems to visit. And, finally, there are some secluded locations you never find in an advertisement in a travel agency. These are places a friend living there would show you, and they are essential for sensing a real spirit of the city, not its pamphlet picturesque image.

Okay, maybe I've taken the analogy a bit too far here. But if you agree to follow it for a second, capabilities would be one of these secluded locations. Introduced with Linux 2.2, they are what really tells if process X can do Y. Yet they are often lost in shadows of traditional Unix privileges, SELinux, eBPF, and many others. By the end of this Core Tech article, you'll know who really sets your limits in a city of Linux.

Listing 1: Permission Checks in Linux Kernels Before 2.2

```
01 extern inline int suser(void)
02 {
03   if (current->euid == 0) {
04     current->flags |= PF_SUPERPRIV;
05     return 1;
06   }
07   return 0;
08 }
09
10 static int inet_create(struct socket *sock, int protocol) {
11   ...
12   switch (sock->type) {
13   case SOCK_RAW:
14     if (!suser())
15     {
16       sk_free(sk);
17       return(-EPERM);
18     }
```

An All-Mighty Root (Actually, Not)

Back in ye olde days, the permission system of Linux was pretty much simple. A user with UID 0 – often called "root" – could do any privileged operation, and he wasn't subject to permission checks. Note it is the UID, not the name, which is important. A user called "val" with UID 0 holds all powers of root user as well.

This "all-or-nothing" approach served well but wasn't very flexible. What if you do not want someone to install new packages or add new users, yet want him to create raw sockets the `ping` command uses? Granting someone a permission to adjust the system date doesn't mean you would be happy if he or she reconfigured Nginx or MySQL on this server.

The `sudo` tool solves this, kind of. You can tell it which command a given user can execute as root, so being able to run `date` doesn't imply a permission to execute `passwd`. But command-level granularity is sometimes too coarse to be useful. If a single command writes files and adjusts dates, `/etc/sudoers` provides no way to restrict the former and grant the latter. This means that you leave your system possibly open to the attack.

Let's look at how the kernel implements permission checks for privileged operations. Listings 1 and 2 show a relevant part of the `inet_create()` function, which is called in response to the `socket(AF_INET, ...)` system call. The listings also show functions that do actual permission checks; they are really defined in a separate file, `linux/sched.h`.

The code in Listing 1 comes from Linux 2.0. It's straightforward to see that it only evaluates if the current process effective UID is zero. However, Linux

Secure Bits

Capabilities are a tricky yet flexible system, and you may be wondering why we still keep root users today. A short answer is it preserves backward compatibility and works well in many cases. A longer answer is that you really don't have to.

Starting with Linux 2.6.26, it is possible to establish a root-less, capabilities-only environment. In this environment, UID 0 is treated no differently from any other UID. As process permissions are really granted as per capabilities since Linux 2.2, establishing such an environment only needs some flags to disable special handling of the UID 0. These flags are commonly known as "secure bits."

Perhaps the most important security bit is `SECBIT_NOROOT`, which disables setting permitted and inherited file capabilities to all-ones, as I described. Two other flags, `SECBIT_KEEP_CAPS` and `SECBIT_NO_SETUID_FIXUP` remove the effect of switching between zero and non-zero UIDs.

All these "base" flags also have companion "locked" flags. A locked flag forbids modifications to the corresponding base flag, and it can't be cleared. This means you can set up a secure bits environment the way you want, lock it, and be confident no process could ever change it. Secure bits are managed with a `prctl(2)` system call, and a PAM module would be an appropriate place to do so.

2.2 is not concerned with the user ID anymore. Instead, it checks for specific flags in the process descriptor. These bit flags are essentially capabilities of the process or, more precisely, a thread. Any capability could be either set or reset. (See the "Secure Bits" box for more information.)

Linux understands a few dozen capabilities now; see `linux/capability.h` [1] or `capabilities(7)` man page [2]. The highest capability's number available (zero-based) is also in `/proc/sys/kernel/cap_last_cap`:

```
cat /proc/sys/kernel/cap_last_cap
37
```

I'd be happy to say any privileged operation has a dedicated capability flag now, but it isn't the case. Some capabilities span several operations. For instance, `CAP_NET_ADMIN` permits one to configure network interfaces, manage firewall rules, and modify routing tables (besides other things). You see the grouping is natural, so when a capability feels coarser than you might expect, it's usually not a problem.

As you may have guessed by now, `CAP_NET_RAW` allows creating a raw (and packet) network socket which is useful for the `ping` command and for sniffing tools such as `tcpdump`.

Capability Sets

You may have noticed that the code in Listing 2 checks capabilities in the `cap_effective` member of the process descriptor. There are a few other `cap_something` members as well because each thread in Linux has several associated capability sets. Effective is, of course, what defines capabilities currently in action. Other sets are used, for example, when a thread does an `execve(2)` system call to execute some new code for which you may want different capabilities.

First, there is the permitted capability set. It contains all capabilities a thread may ever assume – that is, add to any other capability set. If a thread drops a capability from the permitted set, there is no way back, at least until the thread executes the same program.

This brings us to the inheritable capabilities set. As the name implies, these are capabilities that are preserved across the `execve(2)` system call. Inheritable capabilities are automatically added to the permitted set when a program is executed. However, this only applies to privileged processes, which either run as root or execute a setuid binary. For everything else, inheritable capabilities are simply ignored. So, if `ping` had `CAP_NET_RAW` in its inheritable set, and you trick it to run a Python interpreter for you somehow, you still won't be able to create arbitrary raw network sockets. Only `ping` could do it, and it properly restricts the use of this powerful feature to innocent ICMP echo requests.

Listing 2: Permission Checks in Linux Kernels 2.2 and Newer

```
01 extern inline int capable(int cap)
02 {
03   if (cap_raised(current->cap_effective, cap))
04   {
05     current->flags |= PF_SUPERPRIV;
06     return 1;
07   }
08   return 0;
09 }
10
11 static int inet_create(struct socket *sock, int protocol) {
12   ...
13   switch (sock->type) {
14   case SOCK_RAW:
15   if (!capable(CAP_NET_RAW))
16     goto free_and_badperm;
```

This raises a question: How do you execute a privileged helper then? This is, in fact, a common scenario: Consider a network management app. You don't need privileges to fill in stuff like an IP address or a gateway. Yet when you apply these settings, the app calls some helper script (often it is setuid-root) to put the configuration you want in effect.

Before Linux 4.3, there was no straightforward way to do this using capabilities. Now we have the ambient capabilities set. A capability in this set must be both permitted and inheritable (the kernel enforces it automatically), and these capabilities are preserved across `execve(2)` calls in unprivileged programs. When you execute a setuid or a setgid program, the kernel clears ambient capabilities to keep things safe.

A process can also directly change capabilities in the ambient set using `prctl(2)` system call. Keep in mind, however, that everything I described so far applies to `execve(2)` only. Forks are nothing special from the capabilities point of view: Both a parent and a child get a bitwise copy of all capabilities set. It's `execve(2)` that matters as it decides which code a thread will ultimately execute.

Capability Math

Now you have the idea of how the kernel implements thread capabilities, but where do these capabilities come from? Nowadays, they're usually attached to an executable file. Linux stores capabilities in a dedicated extended attribute within the `security` namespace [3]:

```
$ getfattr -m - -d /usr/bin/ping
# file: usr/bin/ping
security.capability=0sAQAAAgAwAA➥
  AAAAAAAAAAAAAAAA=
...
```

Interestingly, there is a dedicated capability, `CAP_SETFCAP`, which grants a permission to set file capabilities. This is a sort of chicken and egg problem, although an "all-mighty root" concept solves it easily.

As with thread capabilities, there are several file capabilities set. Perhaps the most important one is the permitted set. Capabilities in this set are automatically granted when you execute a file, even if they aren't in the inheritable set of a thread doing an `execve(2)` call. So, if an executable file has `CAP_KILL` attached, the process will be able to send signals to arbitrary siblings, even if it doesn't run as root. Note that adding a capability to the file's permitted set isn't enough. You should also set a so-called "effective bit" in the file's capabilities. This bit makes permitted capabilities effective, that is, raised in the effective capabilities set after `execve(2)`.

Files also have an inheritable capabilities set, which is ANDed with the thread inheritable capabilities at `execve(2)` time. This is a way of saying "a thread executing this code never should be granted `CAP_X`." If you know the program is going to adjust the system clock and nothing else, limiting the file's inheritable capabilities set to `CAP_SYS_TIME` would mean dropping any other capability a thread may have gained.

If a process calling `execve(2)` runs as root or the binary itself is setuid-root and has no capabilities attached (Figure 1), both permitted and inheritable file sets are assumed to be all ones (remember they are really just bitmaps). That's how the kernel preserves an all-mighty root illusion in 2017.

If the previous text was too verbose for you, the `capabilities(7)` man page neatly summarizes the rules in just four formulas. Think of a process as doing `execve(2)` and let `P(something)` be capabilities in the respective set. Then, new capabilities, `P'(something)`, are defined as:

```
P'(ambient) = (file is ➥
  privileged) ? 0 : P(ambient)
```

If the file is setuid/setgid-root or has capabilities attached, ambient capabilities are cleared.

```
P'(permitted) = ➥
  (P(inherit.) & F(inherit.)) | ➥
  (F(permitted) & cap_bset) | ➥
  P'(ambient)
```

This one is trickier. Thread inheritable permissions are put in the permitted set if file inheritable permissions don't disable them. Then, the file's permitted capabilities are dropped into the mix, subject to the capability bounding set (see the man page [2] for details). Finally, ambient capabilities are added for non-privileged processes.

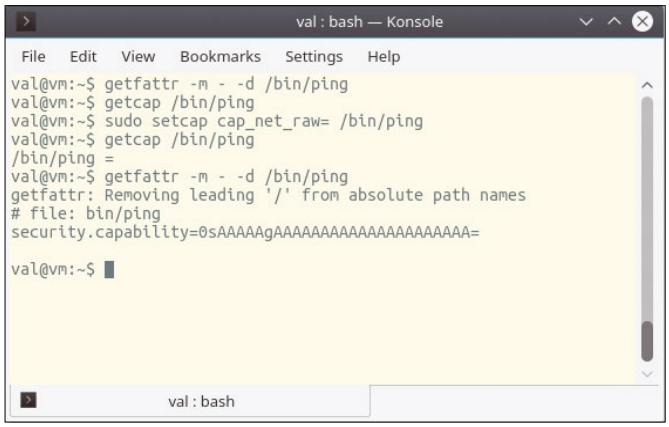


Figure 1: An empty capabilities set is not the same as no capabilities at all, as `getfattr` and `capget` show.

```
P'(effective) = F(effective)
  ? P'(permitted) : P'(ambient)
```

If the file's effective bit is set, permitted capabilities become effective ones. Note that they could never be stricter than `F(permitted)`, `cap_bset` aside. Otherwise, only ambient capabilities are in effect.

Inheritable capabilities remain unchanged during `execve(2)`: `P'(inheritable) = P(inheritable)`. If you are interested in (somewhat mind-bending) implementation details, refer to [4] (also Figure 2).

A Helpless Ping

Enough theory for now – let's go play. As a systems administrator, you are mostly concerned with file capabilities. These are also the most natural way to assign thread capabilities: The kernel does this for you during an `execve(2)` call, following the rules above. In fact, there are only so many ways to change thread capabilities beyond this, even if you are a developer.

Two tools help us here. `setcap` sets file capabilities, while `getcap` reads them back. This is pretty simple. However, capabilities are somewhat convoluted, and so is the syntax of these tools.

`setcap` and `getcap` come via the libcap-something package and are likely already installed on your system, even if it doesn't use capabilities. Among distributions I have on my boxes, Fedora and Arch Linux employ capabilities while Ubuntu doesn't. Yet most distribution kernels now come with capabilities support enabled, so nothing prevents you from "deploying" capabilities on your own.

Consider a simple experiment (Figure 3). I open a terminal on my Kubuntu 16.04 LTS box and type:

```
$ ping 127.0.0.1
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1:
  icmp_seq=1 ttl=64 time=0.095 ms
```

Of course, it works. `ping` is setuid-root binary, so it has no troubles opening raw network sockets:

```
$ ls -l /bin/ping
-rwsr-xr-x 1 root root 44168
  May  8  2014 /bin/ping
```

Now, I do the following:

```
$ sudo setcap cap_net_raw-pe /bin/ping
```

Note that this requires root, as changing file capabilities is obviously a privileged operation. The `CAP_SET_FCAP` capability controls who can do it, but as Ubuntu doesn't use capabilities system-wide, the most straightforward way to get

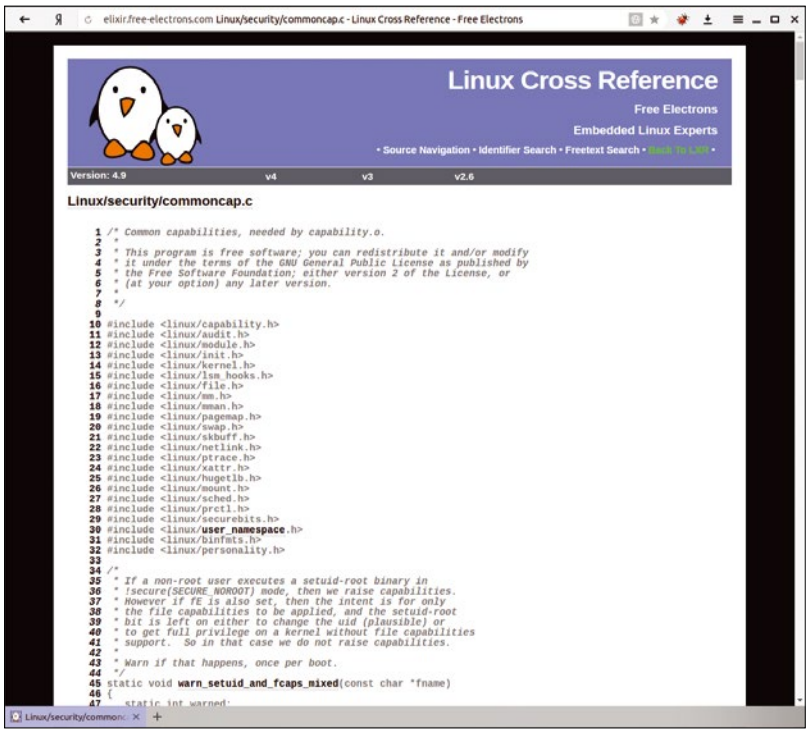


Figure 2: Even if you are a non-programmer, Linux kernel sources are the ultimate authority in how capabilities work.

`CAP_SET_FCAP` in the thread effective set is `sudo`. Now, `ping` becomes hopelessly useless, even if it's still setuid-root:

```
$ ping 127.0.0.1
ping: icmp open socket: Operation not permitted
```

What's happened here? I've withdrawn `CAP_NET_RAW` from the permitted file capabilities and effective capabilities set. The latter is a single bit on file level, so if you tell `setcap` to change effective capabilities, both permitted and inheritable are adjusted accordingly. A companion `getcap` tool will now show file capabilities as empty (Figure 1 again), but it's not the same as no capabilities at all.

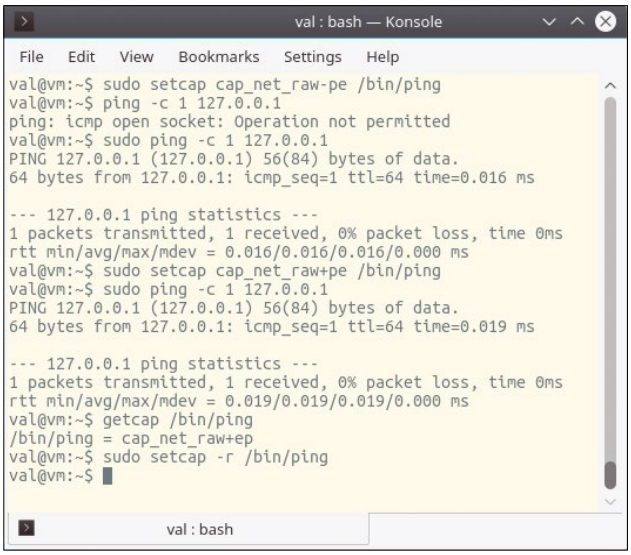


Figure 3: Even if `ping` is setuid-root binary on Ubuntu, revoking a `CAP_NET_RAW` capability renders it perfectly useless.

Info

[1] Linux capabilities list: *http://lxr.free-electrons.com/source/include/uapi/linux/capability.h*

[2] `capabilities(7)` man page: *http://man7.org/linux/man-pages/man7/capabilities.7.html*

[3] "Core Technology: Extended Attributes & POSIX ACLs" by V. Sinitsyn. *Linux Voice*, Issue 30 (September 2016), pp. 94-97.

[4] `cap_bprm_set_creds()` kernel function: *http://lxr.free-electrons.com/source/security/commoncap.c#L492*

[5] `cap_from_text(3)` man page: *https://linux.die.net/man/3/cap_from_text*

Mixing capabilities with setuid/setgid bits isn't a good idea, and you can watch the kernel complaining about it in the logs:

```
$ dmesg | tail
[39034.641585] warning: `/bin/ping' has both setuid-root and effective capabilities. Therefore not raising all capabilities
```

You only get a single instance of this warning regardless of how many times you execute the command. Note, however, that the root user still has all capabilities assigned, even if we've removed one from the file set:

```
$ sudo ping 127.0.0.1
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.095 ms
```

The comments in the kernel says it is done "just to cause the least surprise to an admin."

So, how do you make `ping` operable again? There are two options. The first option is to grant the `ping` command the capability you previously revoked:

```
$ setcap cap_net_raw+ep /bin/ping
```

Now, `getcap` shows it really worked as expected:

```
sudo getcap /bin/ping
/bin/ping = cap_net_raw+ep
```

The second option is to rip `/bin/ping` off file capabilities, just as it was before this experiment. `setcap -r /bin/ping` does just this.

Now you may be wondering what these "cap_net_raw+ep" things are all about. This syntax comes from the `cap_from_text(3)` function, which is a part of the libcap library (not to be confused with libpcap). Each expression starts with a case-insensitive capability name (or "all"), which is followed by a so-called action list. Each action is a combination of the operator (`=`, `+`, or `-`) and a flag (`e`, `p`, `i` for effective, permitted, and inheritable capabilities, respectively). The `=` operator resets the capability in all three sets, whereas two others raise or lower it in a set denoted by the flag. So, `cap_net_raw=` would also do the trick in our experiment, and `cap_net_raw=+i` resets the capability in the permitted and effective set while raising it in the inheritable one. You can find further details in the `cap_from_text(3)` man page [5]. ■■■

Command of the Month: chattr

Capabilities are all about making root a less powerful user. However, there are other ways to limit what root can do on a Linux system.

You probably know how to write-protect a file. Do `chmod ogu-w foo`, and no one will be able to write to it. Do the same for the containing directory, and no one will be able to remove the file. Well, except the root user, obviously.

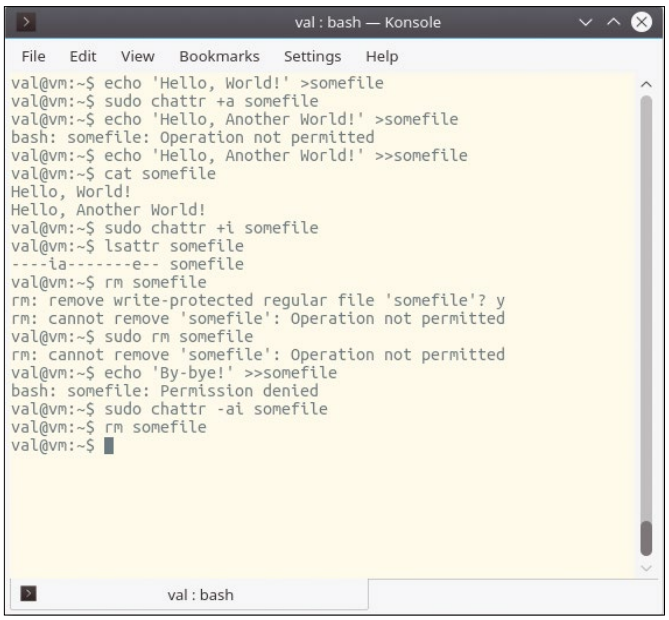


Figure 4: Filesystem attributes aren't strictly about capabilities, yet they also help to do tricky things with your files.

Is there a way to really write-protect a file? Yes, and it comes through a file-attributes mechanism. This mechanism is similar to capabilities in the sense that it's always there, but few people seem to know or care about it. Yet, it makes tricky things possible (Figure 4). Do you want a file you can only append, not overwrite?

```
$ sudo chattr +a foo
$ echo bar >foo
bash: foo: Operation not permitted
$ echo baz >>foo
$ cat foo
baz
```

In a similar fashion, you can make a file immutable with `chattr +i`. Immutable files can't be modified or deleted, and it's impossible to create links to them. This is true even for the root user:

```
$ sudo chattr +i foo
$ sudo rm foo
rm: cannot remove 'foo': Operation not permitted
```

Of course, root or anyone else with `CAP_LINUX_IMMUTABLE` can remove these attributes with `chattr -a` and `chattr -i`, but they are still great to guard against accidental modifications. ■■■

UBUNTU user

MORE UBUNTU!

Can't get enough Ubuntu? We've got a whole lot more!

Ubuntu User is your roadmap to the Ubuntu community. In the pages of **Ubuntu User**, you'll learn about the latest tools, best tricks, and newest developments in the Ubuntu story.

Ubuntu User helps you explore the treasures of open source software within Ubuntu's expansive repositories. We'll bring you exclusive interviews with Ubuntu leaders, keep you current on the exciting Ubuntu community, and answer your most perplexing Ubuntu questions. Learn how to choose a video editor, find the perfect tool to customize your desktop, and configure and manage Ubuntu systems using the best admin tools.

DON'T MISS ANOTHER ISSUE!

HUGE SAVINGS OFF THE NEWSSTAND PRICE!

SUBSCRIBE NOW: SHOP.LINUXNEWMEDIA.COM

FOSSPicks

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

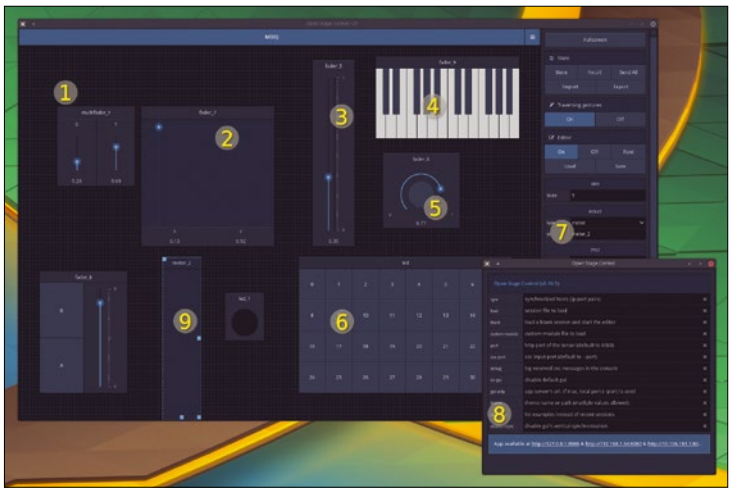
Graham tears himself away from updating Arch Linux to search for the best new free software. BY GRAHAM MORRISON

Hardware automation

Open Stage Control 0.16.5

There's no doubt that the Musical Instrument Digital Interface (MIDI) standard revolutionized music. At its most basic, it enabled musicians and producers to record and play back a performance from a MIDI-equipped keyboard or synthesizer. Then drums, then lights, and then almost everything else. But it was also the harbinger of modern pop music, with its obsession with automation and control over every single aspect of a piece of music. From the swing between each note to painfully autotuned vocals, pop has never been the same. As soon as people got used to dragging notes with a mouse, they wanted to drag everything.

Rants about modern music aside, MIDI is still amazing. It has aged even better than Depeche Mode's *Construction Time Again* ("Everything Counts," excepting) and was born in the same year – 1983. But it was never designed for this crazy interconnected world we find ourselves in. MIDI transfers data at a serial rarity of 31250 bits per second, and the binary package that holds the data has more in common with old modems than modern modulars. This is where the Open Sound Control (OSC) protocol comes in handy. It's an open, modern, hardware-agnostic protocol that's designed for the real-time communication of signals across modern connections. At its core is a URI schema with high-resolution timing and data encapsulation. As with XML for text, OSC allowed creators to design their own specifications and then easily transmit the data encapsulated within OSC. It works brilliantly, whether you're connected via USB or the Internet, and it's really not as complicated as it sounds.



1 **Widgets:** Many different graphical elements can be added to the session. 2 **Faders:** Use buttons, sliders, or XY faders to send data. 3 **Tabbed view:** Several pages can be loaded into a single session. 4 **Keyboard data:** Unlike MIDI, keyboards can send any kind of data you want them to. 5 **Update GUI via OSC:** Send commands to OSC to change the UI and add widgets. 6 **Matrices:** Some widgets can be automatically grouped as matrices. 7 **Editor:** Add, remove, and code widgets to do anything you want. 8 **CLI Options:** Change ports or load share sessions. 9 **Meters:** Various metering widgets represent incoming OSC data.

OSC hasn't yet cracked the hardware synthesizer market, but it's used for all kinds of music and multimedia applications, including Ardour, Pure Data, and Renoise. But it's also perfect for automating lights and other hardware, such as mixers. But, OSC on Linux is otherwise lacking. However, Open Stage Control is one of the best OSC tools I've seen. In one mode, it's an interface designer, letting you add faders, buttons, piano keyboards, VU meters, and many more elements to a blank canvas. With those widgets arranged, you can then define how they send and receive OSC data to your hardware and software. Turn the editor off, and you then use the application as you would a mixer, or a virtual synthesizer, or a remote control. Your arrangement can then be saved as a JavaScript session and loaded again, alongside other templates created by other people. Because Open Stage Control is built with Node.js/Electron, it even runs within Chrome 49 or later. What the application does depends on how you create your templates, and it reminds me of applications like Lemur or TouchOSC for popular tablets, but I've not seen any application on Linux that does this so well, and looks so good. If you're interested in any kind of remote applications, even if that doesn't involve esoteric synthesizer programming, Open Stage Control is definitely worth a few hours of your attention. ■■■

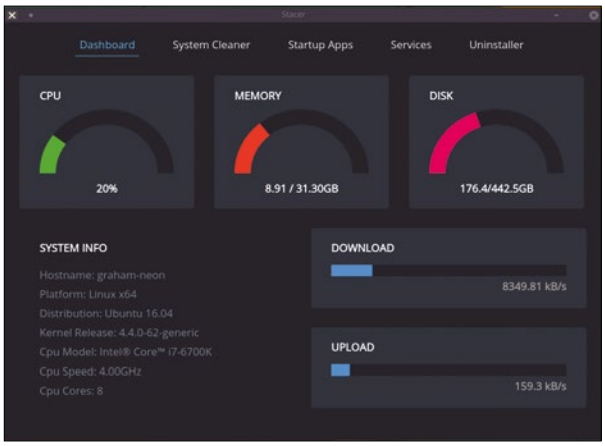
Project Website
http://osc.ammd.net/

System optimizer

Stacer v1.0.4

Optimizing your system used to be a little like making an offering to the gods. You needed to do it with some regularity or your system would suffer strange consequences. This was mostly to do with the cost and limited capabilities of hardware. We didn't have CPU cycles to spare, and storage space was at a premium. The solution was to monitor CPU usage, memory usage, disk space, and networking, and to remove wayward processes, temporary files, and unused packages whenever necessary. Cheap storage and insanely powerful computers have mostly taken the necessity out of this requirement, but keeping an eye on your system is still worthwhile.

Stacer's unique selling point, and it's primary motivation for getting you to monitor your system, is that it looks gorgeous. It has a beautiful user interface that scales smoothly and then dynamically updates to reflect any changes on your system. Start to download something, and the Download meter gently bounces to show the bandwidth being used. CPU, memory, and disk space are similarly monitored, and there are four further pages for options on cleaning out the cruft on your system. System Cleaner will remove old logs and application caches. Startup Apps lets you easily see what's launching when you login. Services shows what systemd has enabled, and Uninstaller is for removing unneeded packages. It all works brilliantly and actually makes you want to refine your system. The only negative is that it's fine-tuned for Ubuntu only, but I could not find anything terribly Ubuntu specific, so there's a good chance it will either work with other distributions anyway or do so after a small update. ■■■



With SSDs being relatively expensive, monitoring free space is now cool again.

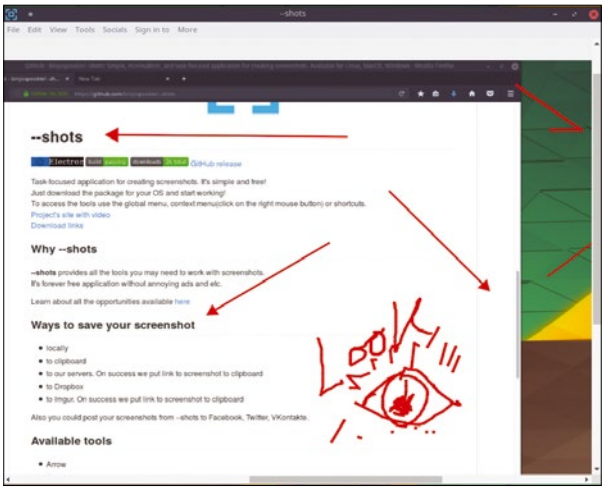
Project Website
https://github.com/oguzhaninan/Stacer

Screenshot tool

Shots 1.2.0

As writers, we may need screenshot tools more than most. Just look at the number of images in this section of the magazine, for example, then multiply that for an entire issue, and then each new issue. But, screenshot tools are useful for everyone, and you probably want a tool that works quickly, is transparent, allows you to easily denote a region, takes a screenshot after a delay, and saves as PNG. As a bonus, if a tool can sensibly increment a filename without a file requester, it gets my vote. If a tool can offer annotations while doing all this, it will win my heart.

No single tool has quite gotten this right. I currently use KDE's Spectacle for its sensible file naming and low profile, along with Hotshots for quick annotations. Hotshots does both, but I find its interface a little too clunky and slow to use. Shots, however, is a new and excellent contender. Not only does it unobtrusively take screenshots – either from the clipboard or from a menu action – it will also let you perform all kinds of manipulation on an image. Most importantly, it will let you quickly annotate an image with text and a draggable arrow, alongside the ability to scribble across your image with a mouse-driven red pen. You can then share or save your image with popular services like Dropbox or Imgur, or on Facebook and Twitter social media accounts. As I'm also a strong believer in doing things yourself, you can automatically upload images to your own server with a small piece of code, which is useful if you commonly work on more than one machine. Shots is a quick, one-step screenshot generation tool that will help you be more productive if screenshots are part of your working life. ■■■



Is it ironic that I had to use a different screenshot tool to take a screenshot of this screenshot tool?

Project Website
https://github.com/binjospookie/--shots

Comic book reader

Peruse 1.2.0

I am Tetsuo.

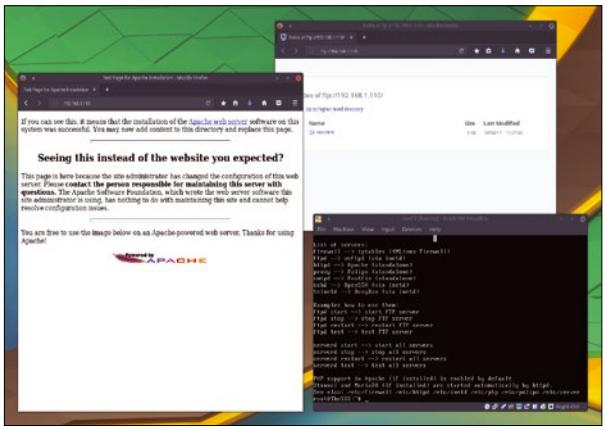
Comic books are brilliant. They're centuries old and yet still feel like one of the most contemporary forms of art, accessible to all, whether that's *The Beano*, *Judge Dredd*, or *Watchmen*. They also seem to have suffered very little from the digital revolution. This is likely because they're tactile and physical. Unlike vinyl records, however, they don't offer any convenience by being digitized. This doesn't mean that digital comics aren't as good, they just offer a different set of advantages. Vast archives of comics are available online, for example, and although printed versions would be wonderful, freely downloadable versions you can archive and peruse at your leisure are great, too.

This is why Peruse is such a useful tool. It works just like a PDF viewer. In fact, it's one of the best PDF and ePub viewers I've used. But it also loads formats commonly used for digital comics, such as bz, cbr, cb7, cbt, cba, as well as weird ones like CHM (I'm not sure if this has ever been used for a comic, although Windows Help is often a joke). After launching, it will scan a predefined set of folders looking for compatible files and present these like a collection of music, as you see in apps like Clementine. Viewing is as you'd expect, with cursor keys or a click left or right to switch between pages. As this is a KDE application, it sits very well within a KDE desktop using a dark theme. The KDE transitions that scroll parts of the UI in and out of view are also lovely, and the UI does a good job of getting out of the way. There's even a *Get New Stuff* panel for downloading comics directly, but this seems linked to the Krita website and doesn't offer much content at the moment. Unfortunately, this is a common problem with KDE's Get New Stuff functionality. But if you've already got access to a collection of comic books, give Peruse a try. ■■■

Project Website
https://peruse.kde.org/

Server suite

TheSSS 21



You can connect to your TheSSS deployment from almost any device, albeit totally insecurely if you insist on using Telnet.

TheSSS is a tiny Linux distribution that can be booted and run off live media. But it's not really like other tiny Linux distributions. Tiny distros have saved my system on more than one occasion, and, because they're tiny, they can usually fit on almost any sized storage medium and boot off all kinds of old hardware. TheSSS is tiny – clocking in as a 67MB ISO – but its size isn't particularly its main selling point. Its selling point is as a quick fix networking tool if you need something low powered and ephemeral to troubleshoot a network or privately connect to something across an insecure network.

When run, you login as `root` with a password of `root` and you then have quick access to several essential tools. SSH as a client and a server are both included and are running. FTP is enabled by typing `ftpd start`, which is very useful for local ad hoc file transfers, and if you need better security, SFTP is included, too. Apache is even ready to run by typing `HTTPD start`, and within moments you've got a local, read-only web server that will run on anything. There's even some help text, and a simple `server` command gives you quick access to the various servers and facilities bundled on the ISO. Behind the scenes, there's a 4MLinux firewall that's using iptables to stop problematic network traffic, a Polipo-based proxy server, and an included Tor anonymizer service. This is a very neat package for such a small footprint, and I can see this having lots of uses, especially if you need quick and dirty network access without wanting to put your phone or your laptop into the firing line of local network conditions. ■■■

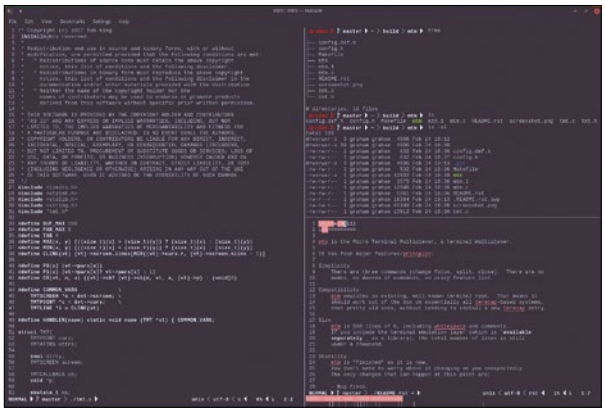
Project Website
http://thesss.4mlinux.com/

Terminal multiplexor

mtm

One thing that's become essential in the midst of this command-line and terminal renaissance is a requirement to use what's known as a terminal multiplexor. This is a rather grandiose term to describe a tool that lets you easily run and control more than one terminal session at a time. Tmux is perhaps the most popular, although screen is installed by default on many distros. Both will let you split a view, move between terminal panes, and create and close new ones as needed. Both are widely used when connecting to remote servers, as they let you detach and resume to a terminal session whenever you connect over something like SSH. Tmux is generally the more advanced because it's easier to expand. There are plugins for saving sessions, navigating GitHub repositories, and saving layouts.

But, tmux gets to be quite processor intensive. This was one of the motivations behind the development of the GPU-accelerated terminal covered in the previous issue (Alacritty). But if you don't have a super powerful GPU handy, another option for faster terminal multiplexing is mtm: "Perhaps the smallest useful terminal multiplexer in the world." It's certainly small. The binary builds quicker than it takes to type `make`, and it loads instantly. There's a different trigger shortcut to both tmux and screen (Ctrl+g), but it does the same thing. After pressing this, you can press another key to perform an action. The minimal nature of mtm means there are only four options: `h` splits the terminal horizontally, `v` splits vertically, `w` removes a terminal, and `/` redraws the terminal. This simplicity is a refreshing change from worrying about keyboard commands, and although it's never going to replace tmux, mtm is a great second option for older hardware or even IoT devices with limited capabilities. ■■■



The mtm tool needs only ncursesw to build, compiles to a diminutive 48k, and will run on almost anything.

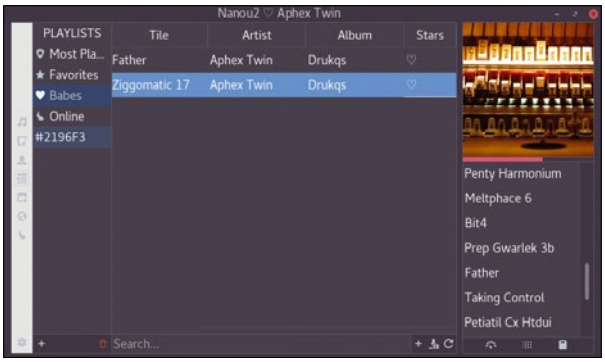
Project Website
https://github.com/deadpixi/mtm

Music player

Babe

Music players seem to be the most prolific type of application you can find on Linux. This must say something about the type of people who use Linux and the kind of things they like doing. It definitely says something about no one agreeing on what the perfect music player should look like when so many developers create their own. Babe, or Qt Babe, or Tiny Qt Babe Music Player as it's called on its GitHub page, is one of those tools. Poor names for audio players aren't unique, but the user interface makes this definitely worth a look, especially if you're from the XMMS old-school. This is because Babe's UI is by default uniquely narrow. It fits perfectly along one side of your screen, showing an album cover at the top and a playlist at the bottom. This vertical block is sprinkled with a small selection of icons for launching a playlist, an album view, adding new music, and controlling playback. These controls and the panels they open look more like the equivalent panes in Clementine or Tomahawk (but not Amarok, never Amarok!), but there is a mode that keeps just the album art on top of your screen, which works really well.

The notification system is also excellent with the regular track pop-up showing you an album cover from the currently playing track but with an extra *Babe it* button. Click on this and you've *hearted* (or *babed*?) the currently playing track. These tracks are then added to one of the many dynamic playlists you can create, all of which are displayed in one of the extra panels that can be opened and closed. Babe is currently in early alpha, and the promised YouTube linking, where online music can be integrated with local playlists, was still a little unpredictable during review. But Babe shows real potential, primarily because it's doing something old in a new way. ■■■



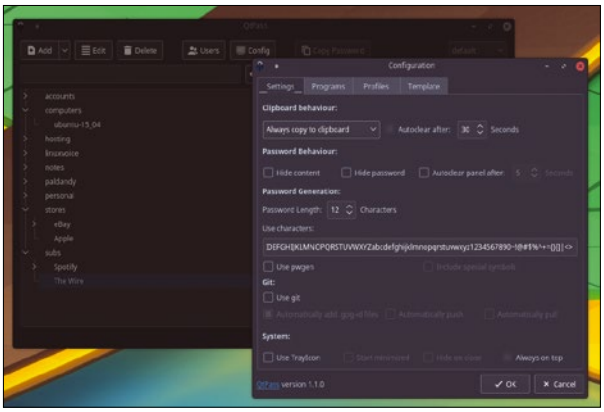
The music player Babe can be made tiny, 200x200 pixels, showing just the album cover and its excellent notifications.

Project Website
https://github.com/milohr/babe-qt

Password manager

pass 1.7

I often mention pass in these pages. This is because it's my preferred password manager and I use it all the time for unlocking important sites, logins, and for storing specific details. But what makes pass so useful, and worth the terrible search hoops you have to jump through because of its name, is that it uses standard GPG to encrypt each password you want to store as a single file within your standard filesystem. You could, for example, have an amazon.co.uk folder and within that have passwords stored within files named after the account(s) you use. If a hacker gains access to your data, they will learn about your account names but they won't be able to access the passwords. The advantage with this approach over a database is that you can see all your passwords, see the organization, and you don't need pass installed even if you need to decrypt (or encrypt) a password. And, if you don't like the suggested naming schema, there's nothing stopping you from obfuscating your own details any way you'd like. The amazon.co.uk folder could just as easily be called KHJGF. Pass does one job, and it does that job well by vastly improving convenience and trust. After two years of development, pass 1.7 adds some fantastic new options. Top of the list is the ability to use extensions, and there are already some excellent examples, including an extension for storing your passwords in Tomb, support for OTP, and an importer for other managers. QR codes can now be generated to help make password access more intuitive, and an awesome new password generator uses `/dev/urandom` directly rather than through the venerable pwgen that was used before. ■■■



Pass works with Chrome and lots of other operating systems. Two of the best UIs, are QtPass and PasswordStore on Android.

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

Keyboard controller browser

Qutebrowser 0.10.0

Like pass (above), Qutebrowser is another of my favorite applications that's just had a serious update. It's a minimal web browser that's controlled entirely by keyboard shortcuts based on those from Vim. This means that if you're proficient at Vim, you're already proficient at Qutebrowser. But even if you're not proficient at Vim, Qutebrowser is easy to use. Type *o* and start entering a search term. Pressing Enter will load the results from DuckDuckGo (by default). Press *f* and each link is shown as a keyboard shortcut, which you can press to open the result without using your mouse, although you can always use your mouse if you need to. Other keys include *i* to enter insert mode, and *:* to enter the interactive internal configuration system.

Qutebrowser has been through a significant development cycle over the past 12 months. This was made possible thanks to a successful crowdfunding campaign used to add support for the vastly superior QtWebEngine that's used by Chromium. I've been using each successive release, and 0.10.0 is the first where I'd recommend defaulting to the new engine over the old QtWebKit. Not only is it vastly quicker and more secure, it brings QuteBrowser into the HTML 5 driven modern age, with many online applications, from Google to GitHub, no longer complaining about your browser version or falling back to HTML modes. But there are lots of other new features, too, including user scripts, support for QtWebKit Reloaded, printing support, and new history commands. After a little practice, QuteBrowser becomes second nature, and it's a brilliant way of navigating the web from your keyboard using a minimal and now fully functioning browser. If you've not tried it – or tried it this time last year – it's definitely time to give the latest release a go. ■■■

Launch the new and awesome QtWebEngine mode with `qutebrowser --backend QtWebEngine`.

Project Website
https://qutebrowser.org

Object Pascal games engine

Castle Game Engine

In the 1990s, Pascal and its cousin Modula-2 were often the languages of choice when studying for a computer science degree. This was because, at the time, Pascal was a good bridge between the pseudo code language of theory and the concepts, functions, and ideas that preempted object-oriented programming. The move away from those old ideas has almost killed off Pascal, but it still exists within Object Pascal (aka Delphi), which is itself a bridge between the old way of doing things and object-oriented programming. Object Pascal survives because its community keeps it going, and this project, the Castle Game Engine, is a good example of that community's tenacity and inventiveness.

The Castle Game Engine helps you to build games with Object Pascal. It can import a huge number of 3D and 2D image formats, but it's primarily built around Scene graph (X3D), which is the successor to the Virtual Reality Modeling Language, commonly used a decade ago to display and interact with 3D environments from your browser. X3D encapsulates these models and interactions and takes the heavy lifting out of the code. Castle can even build and edit these scene graphs at run time to create processing and visualization tools. The games engine also includes plenty of graphical effects, such as shadows, mirrors, and shader effects, and an API to link against other toolkits. The results look fantastic, and perhaps in a nod to VRML, it's

Despite, perhaps, the use of Object Pascal being an anachronism, there are plenty of C modern games developed with the Castle Game Engine.

even possible to run generated games in a web browser. It may seem arcade, but you can learn a lot about games development, and 3D in particular, by looking at these concepts through the lens of a different programming language, which is exactly what Object Pascal excels at being. ■■■

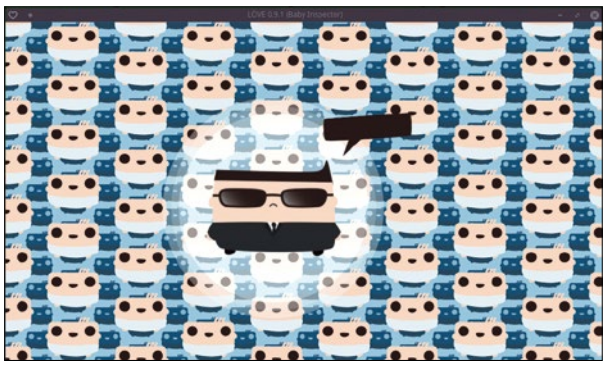
Project Website
https://castle-engine.sourceforge.io/

2D games engine

LÖVE

And here's another game engine! This particular discovery is thanks to the wonderful *Oh My Giraffe* game. If you've not tried the game itself, grab it now. It's a beautiful 2D game that has you playing a hungry giraffe that has to stretch its neck to grab fruit hanging off lots of vines, while at the same time being chased by even hungrier lions. The game is free but not open source, and I'm mentioning it here because it was built using LÖVE, an open source 2D game framework that's almost as awesome as the game itself. The great thing about LÖVE is that it's straightforward and doesn't require too much learning, unlike something like Unity. To play some music, for example, the syntax is as simple as creating a function that executes `sound = love.audio.newSource("music.ogg")`, and other functions are equally easy to understand.

All of this code is wrapped within the Lua programming language. As this language is in common use within the games industry, especially when it comes to add-ons or scripting levels, there's a good chance people will already be familiar with its use, and the demographic is perfect when asking for help. Fortunately, as LÖVE is itself open source, so too are many of the games that have been developed with it. This means you can download the code and the assets, run the game, and take a look at how everything is put together. The



Write a game on Linux and easily move it to Windows, Mac OS X, Android, and iOS.

documentation for LÖVE is a little short on the basics, especially when it comes to making a very simple entry point, which makes the many other projects using LÖVE much more useful. But the results speak for themselves. When you see the animations *Oh My Giraffe* is capable of, and know that the game was created within this same environment, you just want to have a go. ■■■

Project Website
https://love2d.org/

Harness the Web from the Command Line

Not only is it possible to browse the web from the CLI, it's mightily useful, too, as we demonstrate in this article.

BY MIKE SAUNDERS

A few months ago, I was stranded at an airport – my plane was delayed, so I had a couple of hours to kill. Fortunately, it was past beer-o-clock (i.e., 1800 hours), so I headed to the bar and then cracked open my laptop for some Reddit-browsing antics. But there was one big problem: The airport's Internet connection was slow.

I'm talking horrendously slow here – it was like using some 300-baud modem from the 1980s. I pretty much couldn't do anything on the Internet, and I saw other grumbling passengers desperately trying to reconnect multiple times in hope of a teensy bit of better bandwidth. They soon gave up.

Because I'm a turbo-geek, though, I came up with a solution. Somehow, I needed to drastically reduce my bandwidth but still be able to surf the web. I tried disabling images and JavaScript in Firefox; this resulted in a slight improvement, but browsing was still agonizingly slow. And then it struck me: I could SSH into a Raspberry Pi I have at home and browse from there.

Yes, thanks to a couple of rather excellent text-mode browsers, I was soon browsing Reddit threads and Wikipedia pages at lightning speed, despite my terrible connection. The actual job of browsing was being done on the Pi, so only the screen changes were being sent over the SSH connection to my laptop – and those required hardly any bandwidth. Because no images were involved, everything ran buttery smooth.

So, in this tutorial, we're going to look at some ways to navigate the web and grab content at the command line. This may be useful in scenarios such as the one I described; moreover, these techniques could come in useful if you're logged in to a remote server and need to do some browsing from there (e.g., to download something) or even

YouTube from the Command Line

Ever seen a YouTube video that you'd like to keep permanently on your hard drive? Copyright issues aside, there are browser extensions that provide extra links – but often these extensions are bundled with adware or malware. A more elegant (and Linux-y) way is to use a command-line tool: `youtube-dl`.

This lets you specify a YouTube URL and then grab the video data for offline use; it also works with many other video-hosting websites, such as Dailymotion, Streamable, and LiveLeak. The program is available in the package repositories of most popular Linux distros – if you can't find it, though, or you want the very latest version, you can get it from the `youtube-dl` website [1].

Then, you can download a video by providing the URL, for example:

```
youtube-dl https://www.youtube.com/watch?v=rfl9KQb_HVk
```

However, you'll probably want some more control over the exact format of the file that you're downloading. To achieve this, add the `--list-formats` parameter:

```
youtube-dl --list-formats https://↵
  www.youtube.com/watch?v=rfl9KQb_HVk
```

With this, `youtube-dl` probes YouTube to find out which formats are available for the specified video and then lists them with numbers. You can then repeat the command, specifying the appropriate number after the `-f` parameter:

```
youtube-dl --f 32 https://↵
  www.youtube.com/watch?v=rfl9KQb_HVk
```

There are many other options in `youtube-dl` for logging into video sites before downloading, performing post-processing on audio or video (e.g., recoding in a different format), and much more. You can even create templates for output file names – that's especially useful if you want to create compilations of music, for instance. For full details on all the options, check out the program's documentation [2].

want to watch a video (see the "YouTube from the Command Line" box for details). And, of course, browsing the web from the command line wins you oodles of geek points as well. So, let's get started.

The Web in Your Terminal

There are many text-mode web browsers out there, and performance and layout differ significantly. No text-mode browser can compare to a graphical one in terms of advanced layout features, of course – so don't expect a web page's fancy CSS animations and floating div prettiness to shine through inside a terminal window. However, some browsers are much better than others. If you've been in the Unix world for a while, you've probably heard of the Lynx browser; this dates back to 1992, so it's almost as old as the World Wide Web itself. Lynx isn't bad, but there are much better alternatives – we're going to focus on two here.

The first is ELinks [3]. Grab it via your distro's package manager, or on a Debian-based distro, use:

```
sudo apt-get install elinks
```

Now you can simply start it in a terminal by entering the program's name followed by a website:

```
elinks http://elinks.or.cz
```

There you have it: the ELinks website, rendered in plain text, like in Figure 1. In the top-right corner of the window, you'll see the page title, and to navigate through links on the page, use the up and down cursor keys. You can see the address of the currently highlighted link in the status bar at the bottom; use Enter (or the right cursor key) to navigate to the link, and the left cursor key to go back in your history. Try it out by hitting the down cursor key a few times to select the "Linux From Scratch" link, and then press Enter to access it. Keep an eye on the status bar which shows you progress as a page is loading. Once you're done, hit left to return to the ELinks site.

Figure 1 shows ELinks in its default configuration, which is meant to provide basic browsing facilities for running on almost every Unix-like OS, on every terminal type, and over every connection. But we can make it a lot prettier. For starters, we can change the box at the top of the ELinks page from dashes and plus characters into a proper box using better text characters by going to the options in the menu.

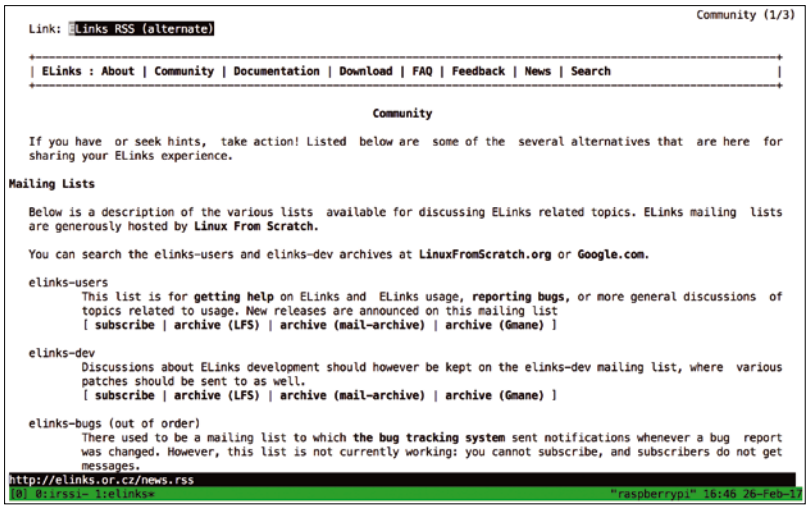


Figure 1: ELinks on its first start – all in black and white. We can easily fix this, though.

But where is the menu? It's not visible by default, but hit F10 to bring it up. (If that doesn't work on your setup for some reason, press Esc followed by Enter). Use the cursor keys to go to the Setup menu and then Terminal options, and then hit Enter. A dialog box will pop up with a bunch of options: Use the down cursor keys and space to select *VT 100 frames* and *265 colors*, and then keep going down to the OK button at the bottom and press Enter.

Suddenly, ELinks is a lot more colorful, and boxes are drawn properly as well. If you open up the menu again, you'll see that it's rendered better,

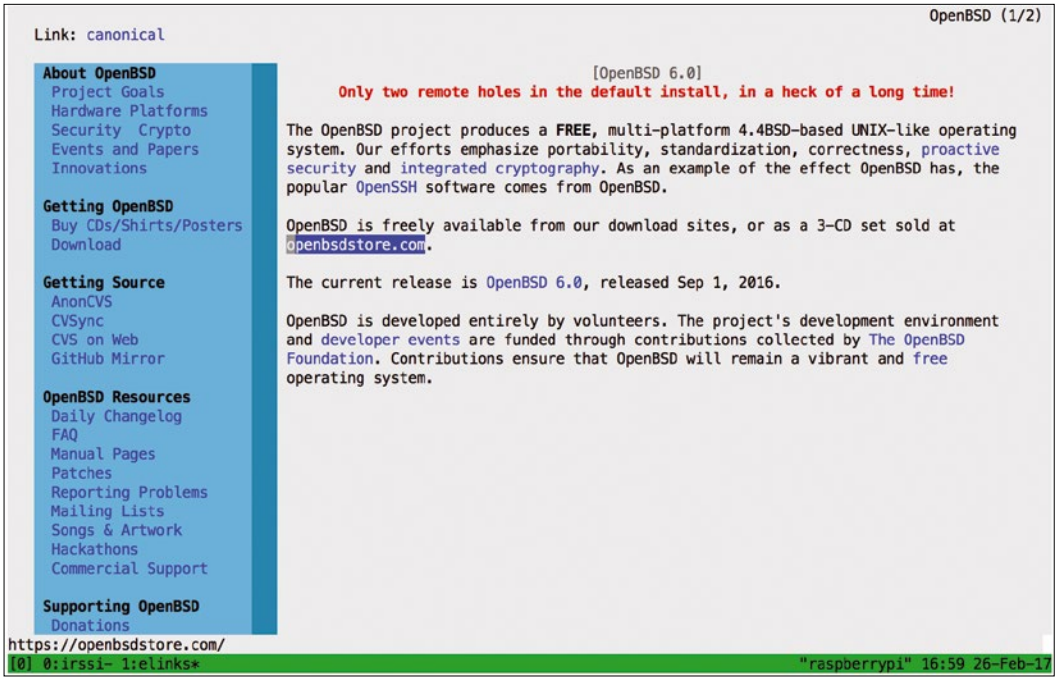


Figure 2: Here's ELinks after a bit of customization, showing the OpenBSD website. Pretty good rendering, for text mode!

too. Note that the settings aren't stored automatically – to do that, go to *Setup | Save options*.

Kicking the Tires

Now let's try ELinks with a slightly more complicated web page. To go to another URL, simply type *g* to open a dialog box and enter an address – try entering `www.openbsd.org`. You'll see the results like in Figure 2, and if you open that OpenBSD website in a regular graphical browser like Firefox, you'll see that ELinks does a pretty decent job of maintaining the layout. Sure, no images are displayed, but the site is more than usable if you just want to get textual information from it.

From here, you can start exploring other websites to see how well they render. As mentioned, it's best to not expect too much, and sites that rely on complicated JavaScript will have problems. But you may be surprised at how well ELinks handles many text-oriented websites. If you just need to pluck some information from a wiki page, it's great.

ELinks has various features that you'd expect in graphical browsers as well: history, bookmarks, the ability to view a page's source code, and more. Check out the View and Tools menus for these – and if you want to download a link rather than visit it, press *d* when the link is highlighted. Finally, when you want to quit and return to the command line, hit *q*.

So that's ELinks – it's great, but it's always a good idea to have a couple of tools in your armory for tasks like this. There's another great text-mode browser that's not quite as user-friendly as ELinks (with its menus and dialog boxes), but actually renders some pages even better: w3m [4]. This was originally developed as a "pager" – that is, a tool for viewing HTML documents on demand, like when they're attached to an email, but it has developed into a dedicated browser in its own right.

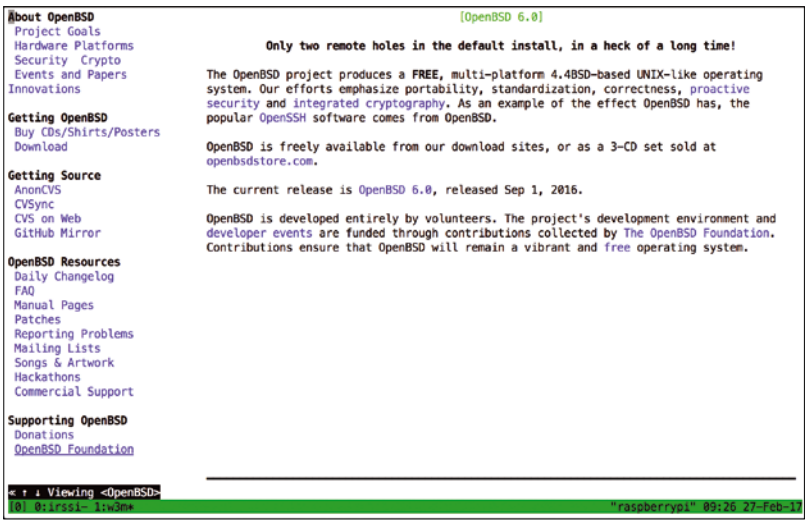


Figure 3: This is w3m, showing the OpenBSD site. It's not quite as true to the original as with ELinks, but w3m is better for some other sites.

To run it, install the *w3m* and, if it's available, the *w3m-img* package as well. On a Debian-based distro (e.g., on a Raspberry Pi running Raspbian), you can use this command:

```
sudo apt-get install w3m w3m-img
```

Now start it by providing a website:

```
w3m www.openbsd.org
```

Here you can see how w3m attempts to render the OpenBSD website, like in Figure 3. (Compare it with how ELinks renders the same site.) You'll note that in its default configuration, w3m is more colorful than ELinks, so you don't need to tweak any settings straightaway. Navigation is quite different, though: You can use the Tab key to move the cursor to the next link in a page or use Shift+Tab to move the cursor to the previous link.

Alternatively, you can use your arrow keys to move the cursor around freely (or H, J, K, and L in vi-style), and once the cursor has landed on a link, tap Enter to follow that link. To go back in your browser history – that is, return to the previous

w3m and Reddit

Reddit is (usually!) a good source for news and discussion about a zillion topics, and the free and open source communities are well represented too. W3m is especially good for browsing Reddit discussions (Figure 4), but if you go to a normal Reddit page (like *www.reddit.com/r/linux/*), you'll notice that the text-mode browser struggles to recreate the layout. Navigate to a comment link and you'll see that you have to scroll past lots of unrelated content before you can see the comments – and then the indentation of them, which is essential to following Reddit discussions, doesn't work properly either.

Fortunately, there's a solution to this. Reddit provides a few alternative views for older or more limited web browsers. Simply add *.mobile* on to the end of a subreddit, and Reddit will send back much simpler HTML – so for the Linux subreddit, go to *www.reddit.com/r/linux/.mobile*. Go to one of the comments links (e.g., *83 comments*), then scroll down, and you'll see that it's much easier to follow discussions.

Reddit has its share of trolls and memes, but some of the discussions are stimulating, and if you're ever stuck in a super low-bandwidth situation, you can keep yourself busy for hours by browsing various subreddits.

page, hit *B* (note the capital letter). And to quit the browser, press *Q*.

To go to another website, press *U*, which opens up a prompt at the bottom of the screen. Simply type in the address and hit Enter. If you're reading a long page, you can use the space bar to jump down a page at a time – this is where w3m acts like a pager, similar to the good old `less` command.

W3m even supports tabbed browsing: Press Shift+T to open the current URL in a separate tab, and then use Shift along with the square bracket keys – [and] – to navigate between tabs. To close a tab without terminating the whole browser, press Ctrl+q. (See the "w3m and Reddit" box for more options.)

Pretty as a Picture

OK, let's now take text-mode browsing to the max: We're going to display images. Yes, this is possible, and no, it ain't pretty. In fact, there aren't many reasons why you'd want to do this, unless your X server has gone down completely but you still need to look at cat pictures on the Internet – but it's fun to try out nonetheless. First of all, you need to grab a package called *caca-utils* – if you can't find it as a standalone package, try installing *libcaca* as well. This is an ASCII-art library – ASCII being plain text – which converts images into the appropriate numbers, letters, and colors for displaying inside a terminal window.

Now restart w3m and navigate to an image file. For this example, we're going to use a JPEG file [5]. W3m will automatically launch a viewer from the *caca-utils* package, and the result should look something like the screenshot in Figure 5. Can you make that out? It's supposed to be a dog, and the white block in the top-left is some text. If you squint you can just about make out that it's a dog – obviously, the text in the image is too low-definition, though.

In the image viewer, try pressing *D* and *A* to turn on dithering and anti-aliasing; your mileage will vary though, depending on the image. If you really, absolutely must see an image in text mode, however, you can always reduce the font size in the terminal and then make the terminal much bigger, so that there are more rows and columns of characters. This provides you with more "pixels," thereby making the image (slightly) easier to recognize. To exit the image viewing and return to w3m, press *q*.

If you want to save an image in w3m rather than view it, press *a* – you'll then be prompted to confirm the filename. (Indeed, you can use the same keybinding to download any link to a file for storing locally.) You can then view the image like so:

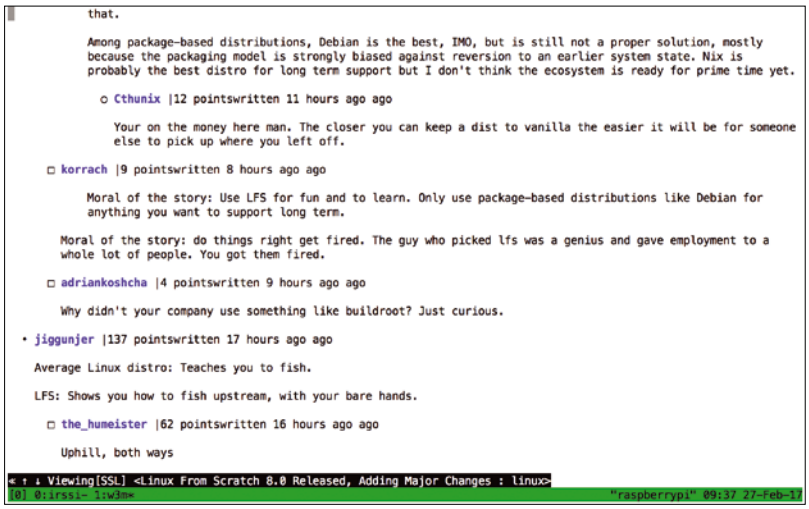


Figure 4: W3m is great for browsing Reddit discussions – just make sure you add ".mobile" to the end of the subreddit URL.

```
cacaview <filename>
```

There are some other utilities in the *caca-tools* package that are worth playing around with as well. Try entering `cacademo` to see what other funky ASCII-art effects are available – use Esc to quit, and then try running it again for a different randomly-selected demo. Another program to try is `cacafire`, which as the name suggests creates a fire effect roaring up from the bottom of the terminal. It's quite impressive to see what's possible with mere text characters – indeed, the `cacafire` program makes a rather cool screen saver when you're accumulating geek points! ■■■

Info

[1] youtube-dl website: *https://rg3.github.io/youtube-dl/*

[2] youtube-dl documentation: *https://github.com/rg3/youtube-dl/blob/master/README.md*

[3] ELinks: *http://elinks.or.cz*

[4] W3m: *http://w3m.sourceforge.net*

[5] JPEG file: *http://i.imgur.com/tpalqEq.jpg*

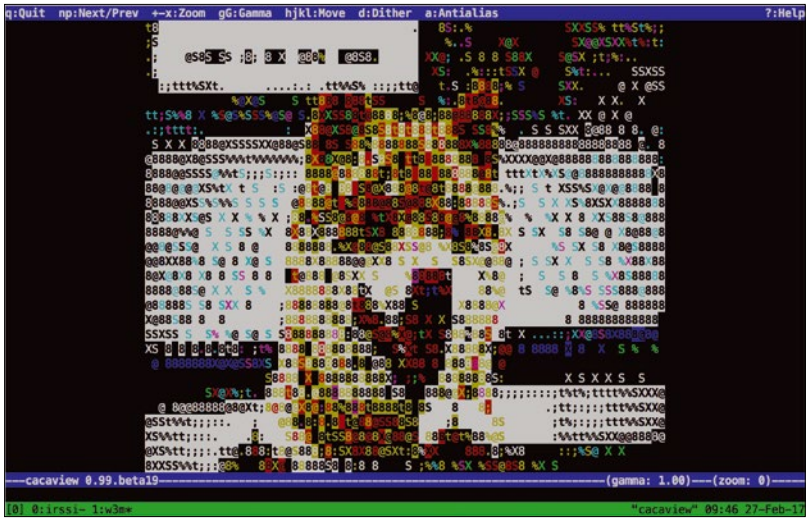


Figure 5: We can even view images in text mode, like this dog from a meme pic. (Can you make it out?)

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*.

OSDC Berlin

Date: May 16–18, 2017

Location: Berlin, Germany

Website: *https://www.netways.de/events/osdc/overview/*

OSDC offers the opportunity to meet open source professionals and insiders and gather and share expertise over three days of presentations, hands-on workshops, and social networking. OSDC aims to "simplify complex IT infrastructures with open source" for experienced administrators and architects.

ISC High Performance

Date: June 18–22, 2017

Location: Frankfurt, Germany

Website: *http://isc-hpc.com/*

The 32nd conference in the ISC High Performance series focuses on HPC technological development, its application in scientific fields, and its adoption in commercial environments. The conference offers 12 HPC topics grouped under three categories: systems, applications, and emerging topics. You can also look forward to ISC tutorials, workshops, and the exhibition.

SUSECON 2017

Date: September 25–29, 2017

Location: Prague, Czech Republic

Website: *http://www.susecon.com*

SUSECON 2017 features exceptional technical content presented by SUSE employees, customers, partners, and community enthusiasts. SUSECON focuses on helping you find ways to build and define your future to provide a flexible and efficient infrastructure. The dynamic and entertaining keynote speeches inspire, inform, and invigorate.

EVENTS

Linux-Infotag	April 22	Augsburg, Germany	https://www.luga.de/start/
Grazer Linux-Tage 2017	April 28–29	Graz, Austria	https://www.linuxtage.at/
Check_MK Conference #3	May 2–4	Munich, Germany	http://mathias-kettner.de/
Linux Presentation Day 2017.1	May 6	Europe-wide in numerous cities	https://linuxday.ch/index.php/en/
LinuxFest Northwest 2017	May 6–7	Bellingham, Washington	https://www.linuxfestnorthwest.org/2017
Open Source Data Center Conference	May 16–18	Berlin, Germany	https://www.netways.de/events/osdc/
DevConf Panamá 2017	May 25-26	Panama City, Panama	https://www.devconfpanama.com//#/
openSUSE Conference I OSDC 2017	May 26-28	Nuremberg, Germany	https://events.opensuse.org/conference/oSC17/
PyConWEB 2017	May 27–28	Munich, Germany	https://pyconweb.com/
ISC High Performance (ISC 2017)	June 18–22	Frankfurt, Germany	http://www.isc-hpc.com/
AnDevCon	July 17–19	Washington, DC	http://www.andevcon.com/
InterDrone	September 6–8	Las Vegas, Nevada	http://www.interdrone.com/
Storage Developer Conference (SDC)	September 11–14	Santa Clara, California	http://www.snia.org/events/storage-developer
SUSECON 2017	September 25–29	Prague, Czech Republic	http://www.susecon.com/
SC17	November 12–17	Denver, Colorado	http://sc17.supercomputing.org/
Linux Presentation Day 2017.2	November 18	Europe-wide in numerous cities	http://www.linux-presentation-day.org/

CALL FOR PAPERS

We are always looking for good articles on Linux and the tools of the Linux environment. Although we will consider any topic, the following themes are of special interest:

- System administration
- Useful tips and tools
- Security, both news and techniques
- Product reviews, especially from real-world experience
- Community news and projects

If you have an idea, send a proposal with an outline, an estimate of the length, a description of your background, and contact information to *edit@linux-magazine.com*.

The technical level of the article should be consistent with what you normally read in *Linux Magazine*. Remember that *Linux Magazine* is read in many countries, and your article may be translated into one of our sister publications. Therefore, it is best to avoid using slang and idioms that might not be understood by all readers.

Be careful when referring to dates or events in the future. Many weeks could pass between your manuscript submission and the final copy reaching the reader's hands. When submitting proposals or manuscripts, please use a subject line in your email message that helps us identify your message as an article proposal. Screenshots and other supporting materials are always welcome.

Additional information is available at:
http://www.linux-magazine.com/contact/write_for_us.

AUTHORS

CONTACT INFO

Editor in Chief
Joe Casad, jcasad@linux-magazine.com

Managing Editor
Rita L Sooby, rsooby@linux-magazine.com

Localization & Translation
Ian Travis

News Editor
Swapnil Bhartiya

Copy Editors
Amber Ankerholz, Amy Pettle

Layout
Dena Friesen, Lori White

Cover Design
Lori White

Cover Image
© alexutemov, 123RF.com

Advertising – North America
Ann Jesse, ajesse@linuxnewmedia.com
phone +1 785 841 8834

Advertising – Europe
Brian Osborn, bosborn@linuxnewmedia.com
phone +49 89 99 34 11 48

Publisher
Brian Osborn, bosborn@linuxnewmedia.com

Marketing Communications
Gwen Clark, gclark@linuxnewmedia.com
Linux New Media USA, LLC
616 Kentucky St.
Lawrence, KS 66044 USA

Customer Service / Subscription
For USA and Canada:
Email: cs@linuxpromagazine.com
Phone: 1-866-247-2802
(Toll Free from the US and Canada)
Fax: 1-785-856-3084

For all other countries:
Email: subs@linux-magazine.com
Phone: +49 89 99 34 11 67

www.linuxpromagazine.com – North America

www.linux-magazine.com – Worldwide

Printed in Germany

Distributed by COMAG Specialist, Tavistock Road, West Drayton, Middlesex, UB7 7QE, United Kingdom

LINUX PRO MAGAZINE (ISSN 1752-9050) is published monthly by Linux New Media USA, LLC, 616 Kentucky St., Lawrence, KS, 66044, USA. Periodicals Postage paid at Lawrence, KS and additional mailing offices. Ride-Along Enclosed. POSTMASTER: Please send address changes to Linux Pro Magazine, 616 Kentucky St., Lawrence, KS 66044, USA.

Published monthly in Europe as Linux Magazine (ISSN 1471-5678) by: Sparkhaus Media GmbH, Zieblandstr. 1, 80799 Munich, Germany.

Approximate On Sale Date

UK / Europe	May 08
USA / Canada	Jun 02
Australia	Jul 03

Issue 199 / June 2017

Simple Regex Language

Next month, we'll show you a kinder, gentler approach to regular expressions.

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

Image © marc hericher, Fotolia.com

DOMAINS | MAIL | **HOSTING** | eCOMMERCE | SERVERS

PERFORMANCE

Level 1 UP TO 2.5 GB RAM
Level 2 UP TO 6 GB RAM
UP TO 9 GB RAM
Level 4 UP TO 15 GB RAM
Level 5 UP TO 19 GB RAM

NEW!

NEXT LEVEL HOSTING!

NEW: Full performance scalability! Customize your web project's performance level yourself, according to your own needs. For example, with increasing traffic from newsletters, blogs, adding an online shop or resource intensive applications, you can easily adjust your site's performance level to match.

✓ Next Level Performance

- **NEW:** 2.5 GB RAM
- **NEW:** Simple scalability with up to 19 GB RAM in just a few clicks
- Performance monitoring in the 1&1 Control Panel
- Change your performance level with no downtime

✓ Next Level Speed

- **NEW:** Faster load times with HTTP/2
- **NEW:** PHP 7.1 + OPcache
- 1&1 CDN
- 24/7 Support

✓ Next Level Security

- Including SSL certificate
- DDoS protection
- Geo-redundancy

Starting at
~~$7.99~~ **$0.99**/month*

1 TRIAL TRY FOR 30 DAYS ★★★★★
1 MONTH FLEXIBLE PAYMENT OPTIONS ★★★★★
1 CALL SPEAK WITH AN EXPERT 24/7 ★★★★★

1 (844) 296-2559

1&1®
1and1.com

*Discount based on 12 month contract for Basic package at $0.99/month, then $7.99/month.

NEW

BigTwin®

The IT Industry's Highest Performing Twin Multi-Node System

Front View

SYS-2028BT-HNR+/HTR+

Rear View

Introducing the First and Only Twin Multi-Node System supporting the highest performance 205 watt CPUs, Intel® Xeon® E5 2600 v3/v4 processor family, full 24 DIMMs of memory, 24 All-Flash NVMe Drives and an industry leading 30% better thermal capacity for maximum performance efficiency and free air cooling (1.05 PUE) plus 96.5% high-efficiency, fully redundant 2200W, 2600W PowerStick- optimized for today and future proofed for the next generation of technology breakthroughs .

Learn more at www.supermicro.com/BigTwin

Intel Inside®. Powerful Productivity Outside.

Use Your QR Code Reader for Videos, White Papers and Brochures