PeerTube: Keep your videos free from YouTube's tricky tracking

debian **11.6** Bullseye

ISSUE 269 APR 2023
LINUX

ENDEAVOUROS 22.12

ISSUE 269 APR 2023
LINUX MAGAZINE

FREE DVD

LINUX
MAGAZINE

ISSUE 269 – APRIL 2023

The **Fediverse**

Social media the FOSS way

- diaspora*
- Mastodon
- PeerTube
- Pixelfed

Easy!Appointments: Manage your meetings and social engagements

Explore the Linux Boot Process

Google Authenticator: Protect your server with multi-factor authentication

SMath: Free MathCAD alternative

ImageMagick: Tweak graphic images at the command line

10 TREMENDOUS FREE TOOLS!

openSUSE Conference

May 26 - 28, 2023

events.opensuse.org

IDENTITY MATTERS

Dear Reader,

The company behind Facebook just announced a new premium subscription model, where you pay for your Facebook account and the payment includes some add-on services that you don't currently receive. I always wondered if a subscription service might be a better model for media giants like Facebook. After all, they do provide a service to their subscribers – maybe asking people to pay directly is better than selling user data and acting as a conduit for targeted advertising.

After the encouraging headline, however, I was disappointed to see that this new subscription-based service does not make your data more private or your online presence less ad-choked. The principal purpose of your for-pay Facebook page is to make sure you are you and not somebody else. The service will provide a "verification badge," showing that you have presented a government ID to ensure that you are who you say you are. The goal is to offer some protection from rogue users pretending to be you.

Corporate social media culture, like the Microsoft lala land that preceded it, occupies a dreamy space where people tell you things and you just nod your head like it is all perfectly reasonable, even when it isn't. If you have to pay extra to ensure that no one is out there pretending to be you (basically, stealing your identity), the other side is that, if you don't pay extra, you can't be sure others aren't out there trying to be you.

I want to be fair. It really does require resources to verify someone's identity and ensure that no one impersonates, so in that sense, the company has a right to pass those costs on to users. But it all sheds light on how faulty the basic service model is in the first place. It is supposed to be against the rules to use a fake name on Facebook, so what this new service says is you have to pay extra for Facebook to enforce its own rules, and if you don't pay the money, Facebook is not responsible for enforcing its own rules.

It might seem like an impossible task to ensure that two billion active users are not impersonating each other. Actually, I believe it is an impossible task, and paying $11.99 per month won't be enough to make it possible. Is it time to admit that Facebook has an impossible business model, rather than letting them climb out of the mess by launching add-on services and claiming the problems are easily solved through consumer choice?

But this is all sour grapes for me, because I'm still disappointed that they would launch a for-pay service and not use the occasion to address the real problem with Facebook, which is all the stalking. Rest assured there are alternatives, though. This month we explore the free social media tools of the Fediverse. The decentralized Fediverse environment will help you with the stalking. As for the impersonation, there's still no universal fix, but check out the article on Mastodon for a workaround that will let you verify your identity to your followers – without paying extra.

Joe Casad,
Editor in Chief

LINUX MAGAZINE

APRIL 2023

ON THE COVER

NEWS

REVIEW

COVER STORIES

IN-DEPTH

The Fediverse

Social media tools connect the world, bringing us the latest news and commentary from politicians, movie stars, community leaders, and remote friends. But the tracking and data mining of the commercial social media platforms has left many users searching for a better option. This month we dive down into the alternative universe for social media users: the Fediverse.

IN-DEPTH

MakerSpace

🐦 **@linux_pro**

f **@linuxpromagazine**

in **Linux Magazine**

m **@linuxmagazine**

LINUXVOICE

debian **11.6**
Bullseye

ISSUE 269 APR 2023
LINUX MAGAZINE

ENDEAVOUROS 22.12

ISSUE 269 APR 2023
LINUX MAGAZINE

TWO TERRIFIC DISTROS
DOUBLE-SIDED DVD!
SEE PAGE 6 FOR DETAILS

EndeavourOS Cassini 22.12 and Debian 11.6 "bullseye"
Two Terrific Distros on a Double-Sided DVD!
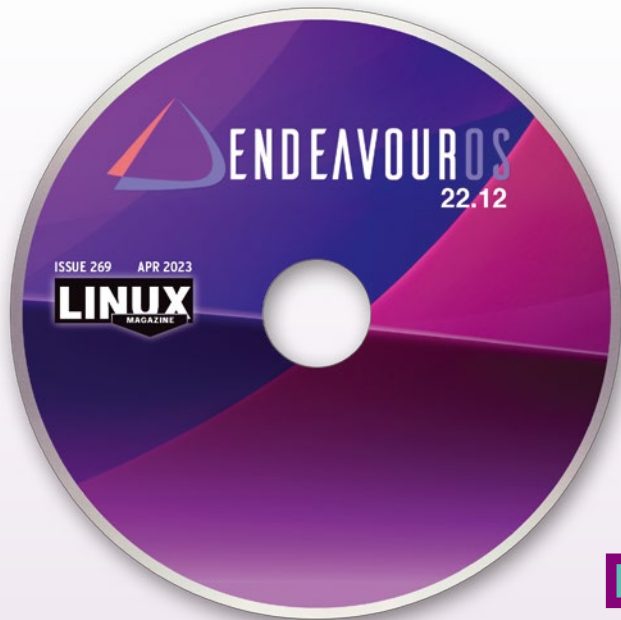
EndeavourOS Cassini 22.12
64-bit

EndeavourOS is to Arch Linux what Ubuntu is to Debian: a desktop-centered derivative that has gained its own following. Unlike Arch, EndeavourOS features an easy-to-use installer, as well as seven desktop environments and five tiling window managers. Like Arch, it is a rolling release rather than a general release distro.

EndeavourOS Cassini 22.12 introduces a number of new features. Among them is a choice of bootloaders, the option to create a new EFI partition, automatic detection of Windows when multibooting, and a redesigned installer, especially for the minimal netinstall. In addition, several desktops have new icon sets, including Plasma, Cinnamon, and Budgie. EndeavourOS also officially supports the Pinebook Pro laptop with hardware and driver support.

If your priority is cutting-edge software in a usable format, EndeavourOS is a prime candidate for your distribution of choice.

Debian 11.6 "bullseye"
64-bit

Debian is among the oldest and largest Linux distributions. Thanks to its stability and security, two-thirds of all the distributions listed on DistroWatch are based on Debian, including Ubuntu and Linux Mint.

Debian 11.6 is the latest point release from the stable repository (aka "bullseye"), the latest general release. Like all Debian releases, it is not the most cutting-edge distro available. However, it is one of the most stable and tested distributions. Like other point releases, the 11.6 release contains security and bug releases, rather than any new features.

Whether you are setting up a server or simply value a reliable installation, Debian 11.6 is a dependable distribution. In particular, if you are having trouble installing Linux, its detailed installer with its extensive hardware support can often provide a solution.

JOIN US AT CLOUD EXPO EUROPE & CLOUD & CYBER SECURITY EXPO

CLOUD EXPO EUROPE

CLOUD & CYBER SECURITY EXPO

CO-LOCATED WITH

BIG DATA & AI WORLD | BARC

DATA CENTRE WORLD

10. – 11. May 2023 Messe Frankfurt
www.cloudexpoeurope.de

NEWS

Updates on technologies, trends, and tools

KDE Plasma 5.27 Released

The next KDE Plasma desktop has arrived and promises to deliver a remarkable experience for one and all.

Most of the newness added to the desktop comes by way of improvements and bug fixes, so you won't find a plethora of new features, but there are some worth noting.

There is a new Flatpak Permissions Settings option in the System Settings tool and a new Welcome to KDE Plasma app that helps guide new users on how to set up online accounts and more.

In this release, however, new features are overshadowed by improvements and bug fixes. Some of the improvements include support for high-resolution scrolling in Wayland; the battery monitor now shows statuses for charging, discharging, and fully charged; Dolphin now selects the next file when you delete a file; and the Discover Flatpak backend is considerably faster.

In addition, the System Settings window decorations now use a more modern "frameless" style (as opposed to the previous tabbed view); the display configuration widget is now found in the System Tray and is inactive when you have only one attached monitor; a multi-monitor revamp has also been undertaken, to make working with multiple displays far more reliable; a Hebrew calendar has been added to the Digital Clock popup; and the KWin Tiling System can be launched with Meta-T.

You can find out more about what can be found in the new update from the official release notes (*https://kde.org/announcements/plasma/5/5.26.90/*). If you want to experience the latest KDE Plasma, your best bet is via KDE Neon testing ISO (*https://neon.kde.org/*).

SparkyLinux 6.6 Available for Installation

SparkyLinux still offers four different desktop editions, LXQt 0.16.0, Xfce 4.16, KDE Plasma 5.20.5, and Openbox 3.6.1, and benefits from all the upgrades found in those environments. The biggest addition to SparkyLinux is the change made to the Sparky Live USB Creator.

With the new release, it is now possible to create live USB installations with persistent storage. That means you can take a version of SparkyLinux with you, wherever you go, that will include any data you've saved to the USB drive.

SparkyLinux 6.6 is powered by the Linux kernel 5.10.166 LTS but does include support for kernel 6.1.

Because SparkyLinux is based on Debian, some of the installed packages may seem out of date, but that is not the case.

SparkyLinux uses the Debian repos, which include stable packages for the installed software. These software packages are the latest stable releases, which are installed from the Debian and Sparky stable repos (as of February 4, 2023). Some of the app versions found in SparkyLinux 6.6 include Firefox 102.7.0.0esr, Thunderbird 102.6.0, and LibreOffice 7.0.4.

You can read more about this point release for SparkyLinux on the official release notes (*https://sparkylinux.org/sparky-6-6/*) and then download an ISO image (*https://sparkylinux.org/download/*) for installation.

Escuelas Linux 8.0 Available

Escuelas Linux is a distribution aimed at the education sector, from preschool to high school, and includes a collection of educational software. This distribution can be installed on either 32-bit or 64-bit hardware and is based on Bohdi Linux.

The list of pre-installed educational software is impressive, with the likes of TuxPaint, GCompris, TuxMath, Scratch, WxMaxima, Geogebra, Kalzium, and many more. Along with the educational software, you'll also find standard packages such as LibreOffice, GIMP, Krita, Inkscape, Firefox, Chrome, Chromium, and more.

As far as the latest upgrade is concerned, the core of the 64-bit release is based on Bodhi Linux 7.0, which is based on Ubuntu 22.04. The 32-bit release is based on Bodhi Linux 6.0, which is based on Debian 11 "Bullseye." The kernel shipped with the 64-bit version of Escuelas 8.0 is 6.0.12, and the 32-bit version ships with kernel 4.19.

For new features, you'll find a brand-new installer, updated applications, and an upgraded Developer Pack, which includes the latest releases of Android Studio, Apache NetBeans, and the Eclipse IDE.

Make sure to read the full release notes for the 25th-anniversary release (*https://news.itsfoss.com/escuelas-linux-8-0-release/*) and then download an ISO for installation (*https://sourceforge.net/projects/escuelaslinux/files/*).

LibreOffice 7.5 Loaded with New Features and Improvements

LibreOffice is the favorite office suite for many Linux users. With the newest release, version 7.5, there are plenty of new features and even some visual refreshing that has gone into the software.

One of the most obvious changes comes by way of a new icon set that is more colorful and vibrant. As well, if you use LibreOffice on a touch-based device, zoom and rotate finally function via multi-touch gestures. And although you won't find major changes to the UI, the subtle changes help to make LibreOffice more modern and professional feeling.

Other changes include several bug fixes for each module, e.g., entering numbers with a single prefix quote in Calc now correctly behaves, based on when to strip the prefix when followed by a number. Calc also now supports DBF Mazovia Encoding for visual FoxPro files, and some conditional formatting conditions are now case-sensitive.

With Writer, you'll find more options for content control, such as plain text and combo box. Now you also can define custom colors for the grammatical mistakes identifier and the spell-checker works with hyperlinks, plus the last modified words now trigger the spell-check when the cursor is moved.

The Writer bookmark module also received a few updates, such as the ability to directly edit a bookmark from the Settings dialog, hidden bookmarks are now correctly presented, bookmarks are now visible within a document, and bookmarks are now readable using screen readers.

The developers also spent a good deal of time improving MS Office compatibility, export features, and the macro programming tool.

To find out more about what's new and improved in LibreOffice 7.5, make sure to read the release notes (*https://wiki.documentfoundation.org/ReleaseNotes/7.5*). You can download the latest version from the LibreOffice site (*https://www.libreoffice.org/*) or wait until 7.5 lands in your distribution's official repositories.

The Next Major Release of Elementary OS Has Arrived

The elementary OS developers have been hard at work crafting version 7 of their open source operating system. With this new release, they've focused on getting users the apps they need, empowering users with new features and settings, and evolving their developer platform.

When you install and boot up elementary OS 7, there are no obvious and major changes to greet you. In fact, the desktop UI looks very much in line with what the team has released in the past, and that's a good thing, because the elementary OS UI is one of the finest on the market.

As for the changes, you'll find more engaging app descriptions in the AppCenter, an easier app update process, better support for sideloading and alternative app stores, and a much more responsive AppCenter.

Version 7 also better supports web apps, making it easier to launch them from the applications menu and even use their own settings (such as privacy controls).

Other new features include the ability to send feedback directly to the developers, a more streamlined installer, and an outstanding Onboarding app that makes it easy for you to configure the UI theme. You can even configure automatic updates for both free and paid apps.

The Mail app has a more modern UI as well as a unified inbox that also supports Microsoft 365 accounts.

The Music app was also completely rewritten from scratch with a focus on more quickly queuing and playing audio files from a local collection.

For more information on the latest release, check out the official release notes (*https://blog.elementary.io/os-7-available-now/*) and then download an ISO (*https://elementary.io/*) image for installation.

KDE Plasma 5.27 Beta Ready for Testing

KDE Plasma 5.27 beta was recently announced (*https://kde.org/announcements/plasma/5/5.26.90/*) and is aimed at testers, developers, and bug hunters. One of the more exciting additions is the new Bigscreen version, which makes the KDE Desktop available for use on televisions.

There is also a new addition to the Display Configuration widget, which now appears active in System Tray by default when you have more than one monitor connected. The Big Multi-Monitor refactor makes working with multiple screens more reliable and gives you fine-grained controls when you have three or more monitors.

The KWin Tiling System is also ready for testing. Using the Meta-T keyboard combination, you can launch the quick tiling features to give you complete control of where your windows are placed.

Another very welcome addition is called Plasma Welcome, which helps you take care of initial setup tasks, such as connecting online accounts. Plasma Welcome is an elegant, simple wizard that should make new users feel right at home on the desktop.

Flatpak gets some added attention in the new release by way of a new System Settings module, called Flatpak Permission Settings app.

This new feature gives you the ability to control the permissions for every installed Flatpak app, including basic permissions (such as Internet connectivity, remote login access, and Bluetooth access), filesystem access, and more.

You can download one of the many live images to test (*https://community.kde.org/Plasma/Live_Images*) KDE Plasma 5.27 beta now.

Netrunner OS 23 Available

Netrunner "Vaporware" version 23 has been made available by Blue Systems (*https://blue-systems.com/*), arriving some two years after the previous milestone release. Unlike Netrunner 21, version 23 migrates to Debian Bullseye, which means it also includes the 5.10.19 Linux kernel.

In addition, Netrunner 23 includes KDE Plasma 5.20, Qt 5.15.2, Firefox 102 ESR, LibreOffice 7.0.4, VLC 3.0.18, Audacious 4.0.5, Thunderbird 102.6.0, GIMP 2.10, and much more. But don't think you'll be getting a stock KDE desktop.

The developers have added a number of customizations to the desktop, such as an overview-like main menu, a unique theme that helps it stand out, simplified System Settings with Plasma Tweaks, a unified look for both KDE and non-KDE applications, GTK apps with standard Kwin borders, Task Manager with expanding icons, Show Desktop hotspot in the lower right corner, and more.

One nice addition is the inclusion of the VirtualBox guest additions, which makes it easier to run the distribution as a virtual machine without having to manually install the software. This means, out of the box, you can resize your Netrunner virtual machine window, without having to first install the guest additions.

Find our more about "Vaporware" on the official Netrunner overview page (*https://www.netrunner.com/netrunner-overview/*) and download an ISO (*https://www.netrunner.com/download/*) today.

New Linux Distribution Built for Gamers

PikaOS (*https://pikaos.ferreo.dev/*) wants to be your new favorite desktop Linux, especially if you're into games. This distribution is similar to what Nobara (*https://nobaraproject.org/*) offers, only instead of using Fedora as its base, it opts for Ubuntu.

PikaOS promises gaming "out of the box," so you don't have to bother with complicated configurations or complex third-party app installation. You'll find the best drivers installed or readily available via the Welcome App and a high level of compatibility for both software and hardware.

PikaOS goes with the XanMod kernel, which implements numerous customized settings to provide ultra-low latency and support for cutting-edge hardware. You'll also find the APX package manager, which is a wrapper for managing APT packages.

One thing to keep in mind, however, is that PikaOS doesn't come with a large selection of pre-installed apps. Even the apps commonly associated with gaming on Linux aren't installed by default.

Apps such as Steam, Lutris, OBS Studio, and more are available to install from the Welcome App. This is the same location where you can apply updates, install hardware drivers, and add multimedia codecs.

PikaOS also includes GDebi, a driver manager, a firmware manager, the Synaptica Package Manager, Firefox, VLC, the Geary email client, the Transmission torrent client, and more. Finally, you can select from five different Gnome layouts to make using Gnome even easier.

To find out more about PikaOS, check out the features page (*https://pikaos. ferreo.dev/#features*) and then download (*https://cdn.ferreo.dev/PikaOS-Gnome-2210-amd64_23.01.18.iso*) an ISO image so you can install the OS on any machine you like.

System76 Beefs Up Popular Pangolin Laptop

System76 recently announced they would soon give their fan-favorite Pangolin laptop a boost with the AMD Ryzen 7 6800U CPU 2.7 up to 4.7 GHz with 8 cores and 16 threads, which includes an AMD Radeon 680M GPU. With the Ryzen 7, users get increased CPU clock speeds, doubled read/write speeds, and faster bandwidth at the tune of 40 percent faster.

The Pangolin chassis is built with magnesium and offers 150 degrees of flexibility for the screen. The display has a 1920x1080 FDH matte with a 144 Hz refresh rate.

Tucked inside the chassis is a battery with 10 percent more life than previous iterations, so you can expect to get roughly 10 hours of productivity from the new configuration.

Other specs include 32GB LPDDR5 RAM at 6400 MHz, 2 x M.2 SSD (PCIe NVMe) with up to 16TB total in storage, 3 USB 3.2 Gen 1 Type-A ports, 1 USB 3.2 Gen 2 Type-C port, an SD card reader, single color backlit keyboard, gigabit ethernet, WiFi 6E, Bluetooth 5.2, video ports (including HDMI 2.0, USB 3.2 Gen 2 Type-C with DisplayPort), full-range speakers and 1 headphone/mic combo jack, a single 1.0 MP 720p HD webcam, and a 70 Wh Li-Ion battery.

As of now, there is no set release date, but the price for the base model will start at $1,299.00. Sign up for updates on the new Pangolin on the System76 teaser page (*https://system76.com/laptops/pangolin-teaser*).

Zack's Kernel News

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

By Zack Brown

Author

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**.

Minimum Version Numbers for Build Tools

Long ago, when this ancient planet was not yet quite so ancient, the Linux kernel refused to support newer versions of the GNU C compiler (GCC). Compiler development proceeded, and Linux would compile only on a specific older version because Linus Torvalds disagreed with some of the new behaviors going into GCC. In other words, it was war.

Today the situation is much different, and the Linux kernel now identifies the minimum version of the tools it requires, rather than the maximum. And today, as Masahiro Yamada pointed out recently, Linux supports GCC version 5.1 and later.

However, Masahiro noted that version 2.23 of GNU Binutils – the assembler, the linker, and other tools essential for building anything with GCC – was now 10 years old and showing its age. Masahiro wanted to raise Linux's minimum supported version to Binutils 2.25, which was released just around the time of GCC 5.1 (seven years ago). He posted a patch to make that change.

It's not a meaningless change! Earlier tools lack support for features Linux actually uses. As a result, the kernel build system needs to check for older tools during compilation. If it finds them, it has to use less efficient workarounds for the more modern features.

Linus said Masahiro's patch seemed sane and accepted it; Nick Desaulniers also liked the patch. There were no dissenting voices.

Masahiro also took the patch into other kernel source trees and suggested that various subsystems and architectures could start to clean up their code. The idea being that they could rip out all the code supporting the older, less full-featured versions of Binutils and simplify the build system somewhat.

This is an ongoing process, and it's not completely free of controversy. If you look at the situation from another angle, the developers are making it harder to compile Linux on absolutely all existing systems. Systems stuck with version 2.23 of GNU Binutils without the ability to upgrade might have a problem compiling newer kernels.

So in fact, it was significant that Masahiro pointed out that Binutils 2.25 was released at around the same time as GCC 5.1. It's very unlikely that anyone with access to GCC 5.1 would be stuck with a significantly older version of Binutils. The decision to require at least GCC 5.1 was made fairly carefully. For example, older versions of GCC actually produced compiler errors at the time Linus decided to update the requirement. So it seemed very unlikely that anyone was actually using the previous minimum version (GCC 4.9) to build kernels. Another reason for making the change to GCC 5.1 was Linus's decision to treat compiler warnings as errors – itself an extremely controversial change at the time – which posed a huge challenge because of all the warnings GCC 4.9 produced in addition to the errors. Rather than fix all those errors and warnings for such an old compiler that probably no one was using, Linus chose to update the minimum version number.

It's always a judgment call. Other operating system projects might insist that older tools must be supported under all circumstances. They would tolerate any amount of bloat in their own kernel source tree rather than force users to upgrade even very old tools. The approach Linus takes is a bit of a compromise with that idea and allows the kernel source tree to continually become cleaner and better organized over time – something very valuable when new developers become interested in contributing.

Intel and ARM Security Issues

Eric Biggers all but accused Intel and ARM of trying to downplay security vulnerabilities in their CPUs. He said that they "recently published documentation that says that no instructions are guaranteed to be constant-time with respect to

their data operands, unless a 'data independent timing' flag […] is set."

Here's the thing about that. An attacker can sometimes gain useful information about something like a password simply by observing how much time it takes your code to reject a given login attempt. If the rejection is quick, it could mean the first characters of the password were wrong. If the rejection is microscopically slower, it could mean the first characters were correct. Then the attacker can keep the earlier characters, make a new guess for the later characters, and try again until it guesses the whole password correctly.

The idea of constant-time execution with respect to data means that the CPU will take the same amount of time to perform a given operation, regardless of whether the data is long or short. This has a direct effect on preventing those exact sort of attacks.

Regarding Intel and ARM's recent statements, Eric continued, "this is a major problem for crypto code, which needs to be constant-time, especially with respect to keys. And since this is a CPU issue, it affects all code running on the CPU. While neither company is treating this as a security disclosure, to me this looks exactly like a CPU vulnerability."

He gave links to the documentation posted by ARM and Intel. That documentation actually said that because the CPUs didn't guarantee constant-time execution by default, they implemented flags that could be set by the operating system to force the CPU to perform constant-time execution.

Eric credited Adam Langley with bringing the whole issue to his attention and asked, "I'm wondering if people are aware of this issue, and whether anyone has any thoughts on whether/where the kernel should be setting these new CPU flags."

Peter Zijlstra heaved a deep sigh, pointing out that these "CPU flags" are actually registers that are specific to a given model of CPU – in other words, the kernel would have to deal with them differently and specifically for each separate CPU model that implemented the constant-time execution flag.

Peter did suggest that it would be better to set and unset the flags as needed by specific processes, rather than

globally set a CPU to always use constant-time execution. That way, for code that didn't need the added security protection, it wouldn't have to tolerate the slowdown.

Arnd Bergmann noticed that some parts of the kernel were already using the flags, and he listed some of the ways it was handled currently. For example, he pointed out some caveats such as "the bit is context switched on kernel entry, so setting the bit in user space does not change the behavior inside of a syscall." He also remarked that currently crypto code inside the kernel did not use the flag, even when available on the CPU.

Jeffrey Walton also remarked, "please make setting/clearing the bit available in userland. Libraries like Botan, Crypto + + and OpenSSL could benefit from it."

Meanwhile Jason A. Donenfeld offered the practical and security-conscious suggestion, "Maybe it should be set unconditionally now, until we figure out how to make it more granular." But he also mused, "I wonder, though, what's the cost of enabling/disabling it? Would we in fact need a kind of lazy-deferred disabling, like we have with kernel_fpu_end()?"

Eric replied, "I'd much prefer it being set unconditionally by default as well, as making everyone (both kernel and userspace) turn it on and off constantly would be a nightmare. Note that Intel's documentation says that CPUs before Ice Lake behave as if DOITM [Data Operand Independent Timing Mode] is always set."

He concluded, "I think the logical approach is to unconditionally set DOITM by default, to fix this CPU bug in Ice Lake and later and just bring things back to the way they were in CPUs before Ice Lake. With that as a baseline, we can then discuss whether it's useful to provide ways to re-enable this CPU bug / 'feature', for people who want to get the performance boost (if one actually exists) of data dependent timing after carefully assessing the risks."

Jason agreed with this, remarking, "It's actually kind of surprising that Intel didn't already do this by default. Sure, maybe the Intel manual never explicitly guaranteed constant time, but a heck of a lot of code relies on that being the case."

Dave also agreed, saying, "I'm in this camp as well. Let's be safe and set it by default."

At this point, a number of these people started discussing implementation details for both Intel and ARM, and the conversation got technical. The email thread ended fairly abruptly – presumably because people started trying out patches.

Removing Support for ICC

Masahiro Yamada posted a patch to remove support for Intel's C compiler from the Linux kernel. He pointed out that nobody had complained when patches were accepted into the kernel build system that explicitly left Intel's C compiler out. For example, "init/Kconfig defines CC_IS_GCC and CC_IS_CLANG but not CC_IS_ICC, and nobody has reported any issue." He concluded, "I guess the Intel Compiler support is broken, and nobody is caring about it."

Linus Torvalds accepted the patch and replied, "I don't think anybody ever really used icc. I can't recall having heard a single peep about icc problems, and I don't think it's because it was *so* good at emulating gcc that nobody ever hit any issues."

Harald Arnesen also pointed out that Intel's compiler binary itself reported that it was a deprecated tool. He ran icc -v and got the message, "remark #10441: The Intel(R) C + + Compiler Classic (ICC) is deprecated and will be removed from product release in the second half of 2023."

Arnd Bergmann affirmed that he also had recently considered removing support for Intel's C compiler. He remarked, "Intel have completely dropped their old codebase and moved to using LLVM, so my guess is that with the current releases it will behave the same as clang."

Dave Hansen, from Intel, also remarked, "I honestly can't remember seeing anyone actually use icc during my entire tenure at Intel. I'll ask around to see if there's any plausible reason to fix this up and keep it. But, I'm not holding my breath."

There was a bit more discussion, and Masahiro updated his patch to avoid removing certain things that related more to GCC and Clang than ICC. But finally it looks as though Intel's compiler will go away.

It was actually a fascinating project and originally represented a significant alternative to GCC – at the time the only such alternative that the kernel could actually use. And given that the kernel developers and the GCC developers have not always seen eye to eye, it seemed like a good idea at the time to have such a fallback available. The days of such conflicts between developer teams, however, seems to have passed. These days the kernel and the GCC developers seem pretty much in lockstep with each other.

Putting the Cart Before the Horse

Rick P. Edgecombe asked Linus Torvalds for some advice about "ecosystem compatibility." In fact, Rick was concerned about features that were becoming very popular elsewhere, but that were not yet included in the Linux kernel. Specifically, he was interested in Intel's hardware-based support for Control-flow Enforcement Technology (CET). This creates a hidden memory stack that prevents certain attacks that might otherwise deduce the memory location containing the next CPU opcodes to execute.

It's a real danger! The kernel's program is just a bunch of opcodes in memory, with a hardware register that continually points to the next one to be executed. Function calls and `goto` statements are always updating where that register points. If attackers can predict that, they might be able to replace those opcodes with their own hostile code.

However, Rick wasn't concerned about CET itself so much as the interactions between other tools and the kernel. Specifically, he said:

"The issues all have a root cause of support for CET in tools spreading widely while kernel support was still in development. This has lead to:
1. Some existing binaries (node.js, PyPy, CRIU) that will break when glibc updates to use the kernel CET APIs.
2. GCC C++ exception stack unwinding code expecting old development versions of the kernel ABI."

Rick pointed out that "once there is kernel support, glibc plans to immediately update in such a way that some existing distro binaries will break against it."

He also said that meanwhile, "The gcc CET support has preceded the kernel changes and the unwinding code assumes things about the kernel shadow stack signal frame ABI that have changed over the course of CET kernel development."

Rick was trying to come up with a way to cushion these future incompatibilities somewhat, and he asked Linus for advice.

Linus, however, gave an elaborate shrug. He said:

"I'm disgusted by glibc being willing to just upgrade and break existing binaries and take the 'you shouldn't upgrade glibc if you have old binaries' approach.

"But hey, I guess that's part for the course for glibc, and there's nothing I can do about that.

"But yes, once people complain, I'll just make sure that old binaries continue to work, and at that point the glibc and tooling people will presumably have to fix their broken situation to get CET at all.

"Because no, the kernel doesn't enable CET if it breaks binaries. That's how we roll."

Florian Weimer replied:

"We've been in this position for years. Every time we use a new system call to implement existing functionality in glibc, some applications break. Mostly due to seccomp filters. They break even if there would be no observable differences for applictions in the way the new system calls would be invoked if the seccomp filter wouldn't block them.

"I proposed a new ENOSYS handshake between userspace and kernel to reduce the amount of breakage (but not all of it). Senior kernel developers rejected it, so we didn't implement it in glibc.

[…]

"Instead, we work with distributions and upstreams to make sure the applications are ready before the next distribution glibc update. Fortunately, there seems to be a pretty broad overlap between seccomp-using applications and applications with frequent, more-or-less mandatory updates, so the transition periods are relatively short. You didn't seem to have noticed, so maybe we aren't doing such a bad job after all."

There was no "official" conclusion during the discussion, although the kernel developers did get into some serious detail about exactly what breakage to expect, who would experience it, and where any potential fixes might go.

Ultimately, Linus probably does mean business in terms of ensuring that user binaries don't break, even if the kernel must accomplish this at the expense of inconveniencing other tools and other groups of developers. Breaking existing binaries and allowing known security holes to persist in the kernel are probably two of the least tolerated things in all of kernel development. Linus would rather accept a huge loss of efficiency and prune off whole branches of kernel features rather than tolerate a security bug or break an existing user binary. ∎∎∎

A return to in-person attendance at FOSDEM 2023

Back on Track

After two years online, crowds of FLOSS supporters returned to Brussels in early February for FOSDEM 2023. Reminiscent of a class reunion, the FOSDEM keynote program was jam packed with goodies. *By Kristian Kißling*

At FOSDEM 2023 [1] in Brussels, FOMO (the fear of missing out) was the buzz word, not only for people who did not make it to the event, but also for the conference attendees. Although the event in the Belgian capital attracted a slightly smaller audience than in years prior to the COVID-19 pandemic, visitors repeatedly turned to live-streams because the conference rooms hosting popular topics quickly filled up.

The First Instance

A particularly large number of comments and FOMO statements appeared on Mastodon, where the *#Fosdem* topic was very popular for some time. Thanks to Elon Musk, Mastodon is currently experiencing a flurry of activity. GitHub developer Kris Nóva, who was unexpectedly dragged into the Mastodon universe last year, vividly described how lively things were. Her plan to build a separate Mastodon instance, Hachyderm [2], for friends took off with thousands of new users gradually migrating to the instance.

Upon reaching 30,000 accounts, the quite powerful hardware in Nóva's basement broke down. It was time to move the service to a cloud, and Nóva chose German provider Hetzner. The data protection regulations (hashtag *#GDPRforever*) were probably also a decisive factor in the move, along with Nóva and her supporting developers needing to field some of the inevitable questions about the general handling of a privately managed Twitter alternative. Nóva pointed out in her keynote that this is where the pitfalls lurk. If you are thinking about setting up a

Mastodon instance, you will definitely want to watch the video [3].

Not only does a Mastodon instance above a certain size need a reliable technical infrastructure, it also faces legal pitfalls. For example, instance operators need to keep the massive amounts of private data secure, which means having volunteers or paid staff moderate the posts. Operators also need to be careful about which other Mastodon instances from the Fediverse they integrate into their platform and which ones they don't. In addition, they also need to consider how to handle corporate accounts [4]. Last but not least, operating a project like this can quickly consume finances. According to Nóva, technical operations of her instance alone incur monthly costs of around EUR1,000.

The solution, according to Nóva, is the newly established Nivenly Foundation [5]. The instance's operators will be looking to collect membership fees through the foundation in the future to refinance operations. This seems to make sense: Where selling user data and advertising cannot be the business model, you need to find alternatives to operate a volunteer project sustainably.

New Energy

Sustainability also played a central role at FOSDEM in another context: the avoidance of fossil fuels and the battle against climate change. The organizers assigned a separate energy track to this pressing problem, along with space in the main stage program.

Felix Rehmann's research project at the Technical University of Berlin, for

example, analyzed the software of 180 research projects that aim to determine, control, or optimize a building's energy requirements. Emissions from buildings are some of the biggest drivers of climate change. The results are exciting.

While most of the 180 projects use at least one open source component, only a few of the mostly publicly funded projects (about 3%) plan to release their software. The reasons range from ethical to technical to cultural concerns. Often, projects simply expire after some time, and the software becomes orphaned. The focus on users is often missing, the licensing situation remains unclear, or the developers are simply afraid of embarrassing themselves. Some researchers also keep their software proprietary for commercial reasons. Despite all these problems, Rehmann is convinced that it would be better to use more open source software in the field, such as CAD programs or simulation software for renovations.

Space, Road, and Data Center

Even NASA made it to FOSDEM (Figure 1). NASA is also looking to use its data to help in the fight against climate change, for example, in the scope of the POWER project [6]. Astronomer Steve Crawford explained that the US space agency is currently opening up to open science, increasingly adopting open APIs, and maintaining open datasets and open access models. However, this is not entirely voluntary. According to a US government decision, starting in

2026 all publicly funded research must also be immediately available to the general public.

Another topic at the conference dealt with electric cars. Vehicle-to-Grid (V2G) pursues the idea of turning electric car users into decentralized electricity producers. The idea is for electric car users to feed the electricity generated from wind or solar energy for their electric vehicle into the power grid in exchange for money. Products to help people do this are available from commercial providers, but they usually look to sell a complete solution. V2G Liberty [7] wants to offer an alternative. Nicolas Höning presented this project in his keynote, along with the setup that was developed. This includes Home Assistant [8], Nextcloud [9], and the LF Energy Foundation-funded Flex-Measures [10], among others.

Höning also explained the potential pitfalls and gave examples of best practices. One thing is clear: The less owners move their cars, the more electricity they can generate. In the ideal case, up to EU10 per day can be earned with vehicles converted in this way. The project is still in its infancy, but that has never deterred open source enthusiasts.

Finally, Parul Singh and Kaiyi Liu presented CO_2-sensitive scheduling for Kubernetes. Admins feed the rules for this to the container platform in the usual declarative style. Kubernetes Efficient Power Level Exporter (Kepler) [11] lets you move pods and their workloads automatically to other areas of the world based on their power consumption. Ideally, CO_2-neutral electricity will then be used there. The containers' energy requirements are determined by eBPF programs, although obtaining the information on the types of electricity is probably not entirely trivial and sometimes also costs money. A software development kit for the Green Software Foundation is also expected to emerge from the project in the end.

Classics

As always, classic programming and admin topics abounded. Once again, people jostled for the best seats at the front of the development rooms for popular programming languages. There were various admin-centric tracks (e.g., on continuous integration and containers) as well as the obligatory kernel track, where the developers of the free operating system kernel presented their pet projects.

On the distributions track, Eric Curtin talked about Fedora Asahi Remix [12]. The Fedora-based distribution for Apple hardware runs on ARM's M1 and M2 processors. Curtin cited a lack of good ARM devices supported by Linux as his motivation. Upstream, the Remix relies on Asahi Linux and offers two kernels of its own: a more stable one and an edge kernel with newer developments. Fedora will be looking to store as many of the software packages they used as possible in Fedora's normal repositories, Curtin emphasized.

Same Procedure …

All told, the familiar FOSDEM feeling quickly set in, with all its advantages and disadvantages. As usual, the speakers in some rooms struggled with the PA (poor sound quality), and long queues formed in front of the coffee stands and food trucks. But the conference scored points with its sheer mass of open source topics, Belgian beer in the cafeteria, and great discussions. In other words: I'll be back … next year! ▪▪▪

Info

[1] FOSDEM: *https://fosdem.org/2023/*

[2] Hachyderm.io: *https://hachyderm.io*

[3] FOSDEM videos: *https://video.fosdem.org*

[4] Account types at Hachyderm: *https://community.hachyderm.io/docs/account-types/*

[5] Nivenly: *https://nivenly.org/about/*

[6] NASA's POWER project: *https://power.larc.nasa.gov/*

[7] V2G Liberty: *https://v2g-liberty.eu/*

[8] Home Assistant: *https://www.home-assistant.io*

[9] Nextcloud: *https://nextcloud.com*

[10] FlexMeasures: *https://flexmeasures.io*

[11] Kepler: *https://github.com/sustainable-computing-io/kepler*

[12] Fedora Asahi Remix: *https://fedoraproject.org/wiki/SIGs/Asahi*

Figure 1: Even a NASA astronomer made it to FOSDEM and spoke about open source in space, among other things.

Introduction to the Fediverse

Social Time

Do you have to you give up your privacy to enjoy access to social media? The makers of the Fediverse say no.

By Joe Casad

As this magazine goes to press, Twitter has just announced that it is closing its API – a radical move that will limit access from outside applications. Some form of API access might be possible, if you pay for it, but as of now, the terms are unclear, and anyway, the move is likely to cause still more restlessness from the Linux community, who have never had much patience for closed APIs and are likely to abandon the platform in even greater numbers. Where will they go?

Social networking and the new generation of users who live on it have changed the Internet. Your thoughts, your experiences, your pictures, your politics – it all goes on the Internet now through your platform of choice: Life events on Facebook; links and short opinions on Twitter; videos on YouTube; photos on Instagram. The whole world is connected, but many users, including many who have never heard of Linux or free software, have already begun to ask whether it is all worth the price.

"What do you mean," the companies tell us. "Our services are all free." But it all depends on what you mean by *free*. Commercial social media services earn billions selling knowledge of their users' opinions and habits. To get on the site, you need to click a box that signs away your privacy and, in some cases, even signs away your ownership of your own words and images.

The whole concept of commercial social media as it is practiced today is decidedly at odds with the values of the Free Software community. Former Free Software Foundation attorney Eben Moglen once referred to social media giant Facebook as a "man-in-the-middle attack," adding. "The point is that by sharing with our actual friends through a web intermediary who can store and mine everything, we harm people by destroying their privacy for them. It's not the sharing that's bad, it's the technological design of giving it all to someone in the middle. That is at once outstandingly stupid and overwhelmingly dangerous." [1]

But over the past several years, while the social media giants were erecting their billion dollar empires, another construction project was quietly underway. Now, with recent controversies at Twitter and growing distrust for Facebook, the work of these Free Software pioneers has made a grand entrance onto the scene. Today a new group of tools threatens to upend the social media orthodoxy. Welcome to the Fediverse [2].

Free Speech with Free Tools

The Fediverse is a collection of servers running social media tools that offer access in a decentralized format without the spying. There is no central authority wielding power over the whole system. Servers operate independently, setting their own rules and managing the services as they see fit. The Fediverse is built on the philosophy that tools should work together, so if you have an account on one, you can view content from the others. In other words, you have an *account* on one server but an *identity* across the whole federated network. As you will learn in this article, this goal of interoperability has been largely successful, although a few links are still needed to complete the chain.

This month you'll learn about some leading tools of the Fediverse, including:

- Mastodon – a microblogging platform that lets you send links and short messages (like Twitter)
- diaspora* – a macroblogging tool, like Facebook
- PixelFed – a photosharing tool, similar to Instagram
- PeerTube – a YouTube-like tool for sharing videos

Tools such as Mastodon and diaspora* are receiving a majority of the attention now, perhaps because they fit so neatly into the niches of the conventional platforms they replace, but, as is often the case with Free Software, users have an abundance of choice. Other tools of the Fediverse [3] include:

- Misskey – a microblogging tool with some extra features, such as a calendar, a chat service, and animated text
- Hubzilla – a multi-purpose macroblogging platform that calls itself a "home for nomads and power users"
- Pleroma – a microblogging service tailored for low-resource systems like Raspberry Pi
- Friendica – an early social media platform that supports multiple protocols for strong connectivity with other Fediverse utilities
- Funkwhale – a music-sharing platform that also supports "socializing around music and discovering new content"
- GNU Social – the GNU project's entry into the search for an alternative microblogging platform

Nostr: Even More Decentralized

The tools of the Fediverse don't have a single point of control like Facebook or Twitter, but you still have to set up an account on a server, and whoever owns that server does have some control over the instance of the service you are using. The Nostr protocol [4] provides an approach that is even more decentralized. The goal of Nostr is to provide a "censorship-resistant global social network." Instead of relying on centralized servers in the classic sense, Nostr depends on relays, very simple components that do not participate in the identity management system and do nothing but forward messages.

The Nostr GitHub page describes the scenario as follows, "Everybody runs a client. It can be a native client, a web client, etc. To publish something, you write a post, sign it with your key and send it to multiple relays (servers hosted by someone else, or yourself). To get updates from other people, you ask multiple relays if they know anything about these other people. Anyone can run a relay. A relay is very simple and dumb. It does nothing besides accepting posts from some people and forwarding to others. Relays don't have to be trusted. Signatures are verified on the client side." [5]

Figure 1: Mastodon is the most popular Fediverse platform, and its popularity is growing with the recent changes at Twitter.

The popularity of the Fediverse tools varies widely. Mastodon (Figure 1) is thought to have more than 6 million accounts and 4 million active users. The smallest platforms might have only a couple thousand accounts. But new attention on the Fediverse could accelerate migration in the years to come.

It is worth noting that the concept of federated servers used with the Fediverse is only one approach to the goal of decentralized social media (see the box entitled "Nostr: Even More Decentralized").

Tools and Rules

Decentralization is one of the principal features of the Fediverse, and it requires a bit of explanation. A tool such as Mastodon is an open source utility that anyone is free to download and implement. The Mastodon community is not all one single user base. Instead, several different organizations act as Mastodon providers (Figure 2). Each provider makes its own rules and manages its own user base. To get started with Mastodon you can sign up with one of the existing providers, or you can set up a Mastodon server and become a provider yourself. Linux New Media, the publisher of this magazine, has an account with the Mastodon provider called fosstodon. In general, Fediverse providers support interaction with other Fediverse providers, so you aren't locked in to a single garden as you are with many of the commercial social media tools, however, the first step is to set up an account.

This independence of the individual providers means that the Fediverse tools aren't controlled by a single company. On the other hand, this diversity means there is no uniformity in how the providers manage their space and how they envision their mission. A server might serve a specific community of interest or political cause. Whoever owns the server is free to moderate the content as they wish. For instance, the veganism.social Mastodon server focuses on

vegans and offers "anti-speciesist moderation by animal rights activists." On the other hand, Truth Social, the social network created by Donald Trump and supported by users with MAGA political leanings, is also based on Mastodon.

Protocols

The secret to the interoperability of the Fediverse tools is that they share common protocols. A close look at the supported protocols will give you a better understanding of what fits with what – and which tools might work best for your situation. For historical reasons, or just to improve versatility, some tools support more than one protocol.

The most popular Fediverse communication protocol is ActivityPub. ActivityPub, which is based on the earlier pump.io and StatusNet protocols, is a standard of the World Wide Web Consortium. Mastodon, PeerTube, PixelFed, Friendica, and other important Fediverse services use ActivityPub.

Diaspora*, which was developed independently and earlier than many of the ActivityPub applications, has its own protocol that isn't universally compatible with the others, although Friendica and Hubzilla have implemented support for it. One benefit of diaspora* is that it has implemented the necessary extensions to interact with a number of commercial social media services. You can propagate posts to Tumblr and WordPress, and you can even embed YouTube and Vimeo videos in content pages. In the past, diaspora* has had some support for propagating posts to Twitter, but that support might be ending with Twitter closing the API.

GNU Social, another one of the early entries, uses the OStatus protocol, which is also supported by Friendica. Hubzilla, like Friendica, places a premium on interoperability and offers built-in support for both diaspora* and the ActivityPub services. Hubzilla also supports the Zot protocol, which allows it to connect with the Zap macroblogging service.
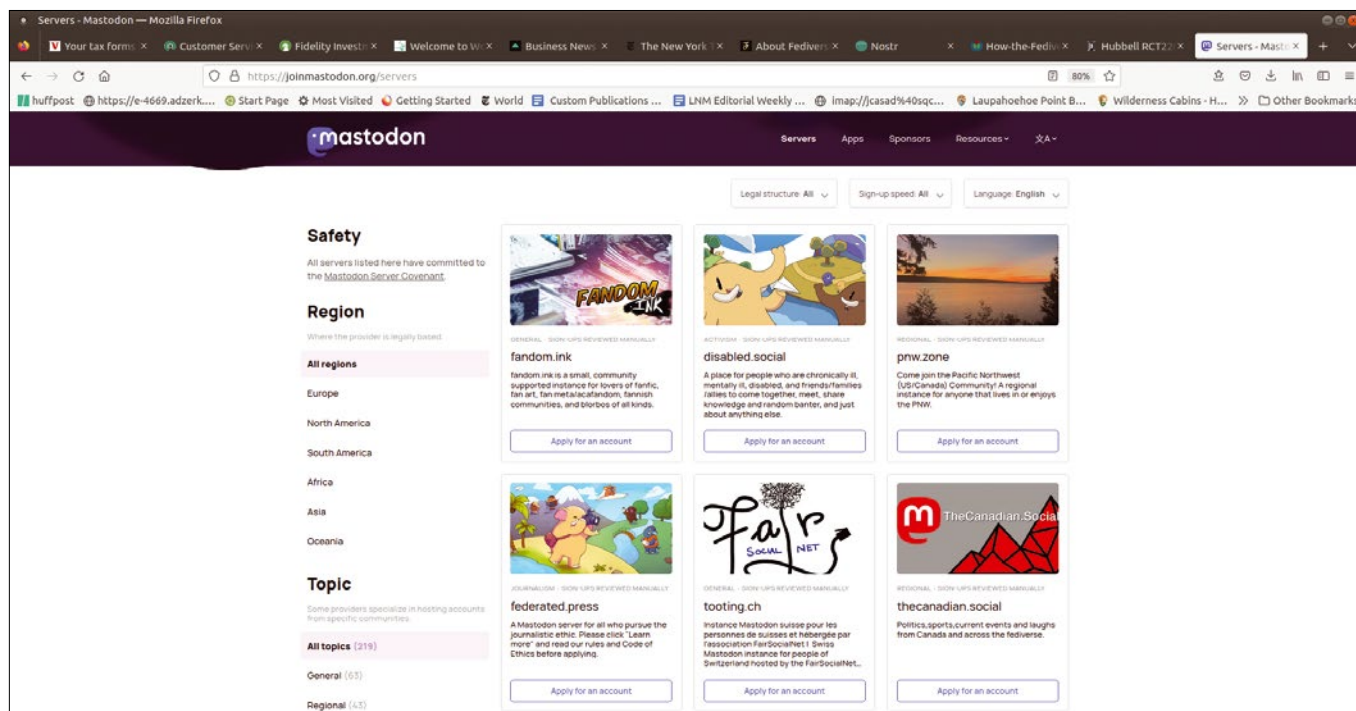

Figure 2: Shop for the server that will host your account at the Mastodon website.

The differing protocols mean that *all* the services in the Fediverse aren't exactly federated, in that they can't all directly connect to each other. In general, you can think of the ActivityPub services as one collection and the orbit around the popular diaspora* service as another group (Figure 3). The versatile Friendica and Hubzilla fit into both camps. GNU Social's OStatus protocol is actually a precursor to ActivityPub and at one point offered a degree of compatiblity. Mastodon supported GNU Social until version 3 but has since discontinued it.

Keep in mind that compatibility among the different services is only part of the equation. In many cases, users are more interested in reaching other users who are within *the same* tool community. For instance, the Mastodon community is a complete ecosystem where millions of users interact every day – they don't measure their value strictly on the ability to reach other services. The Mastodon community is a federation all by itself, with many servers acting independently yet offering

compatibility. Their mission isn't just to offer a backdoor to reach Twitter. They would rather have Twitter users give up their Twitter accounts and come join Mastodon.

Service and Server

Free software is all about choice, and the Fediverse is no exception. An abundance of tools means you'll find the right tool for an abundance of situations, and the freedom to build your own server and target it to your own community will help to spread the influence of the Fediverse throughout the world of Internet users.

Will the Fediverse conquer the world and displace the social media giants? That all depends on whether the world starts watching. If the world's rock idols, politicians, movie stars, and corporate accounts get fed up with Twitter and Facebook and switch to the Fediverse, they will no doubt bring many users with them. If journalists start watching Mastodon accounts the way they watch Twitter accounts now, and treat every post as a quotable news story, the tide will certainly start to turn. But the famous people quoted in the news are only a small fraction of the users on social media. Most people just use social media as a way to communicate with their friends, and it is very easy to imagine circles of friends gradually making the switch to free tools that will protect their privacy and keep them free from data mining and targeted advertising.

Read on for a look at some leading tools of Fediverse. And remember: This issue is just a snapshot of where we are right now. Things change fast with Free Software, as developers adapt to new ideas and changing times. If you don't find what you need today, keep checking! ∎∎∎
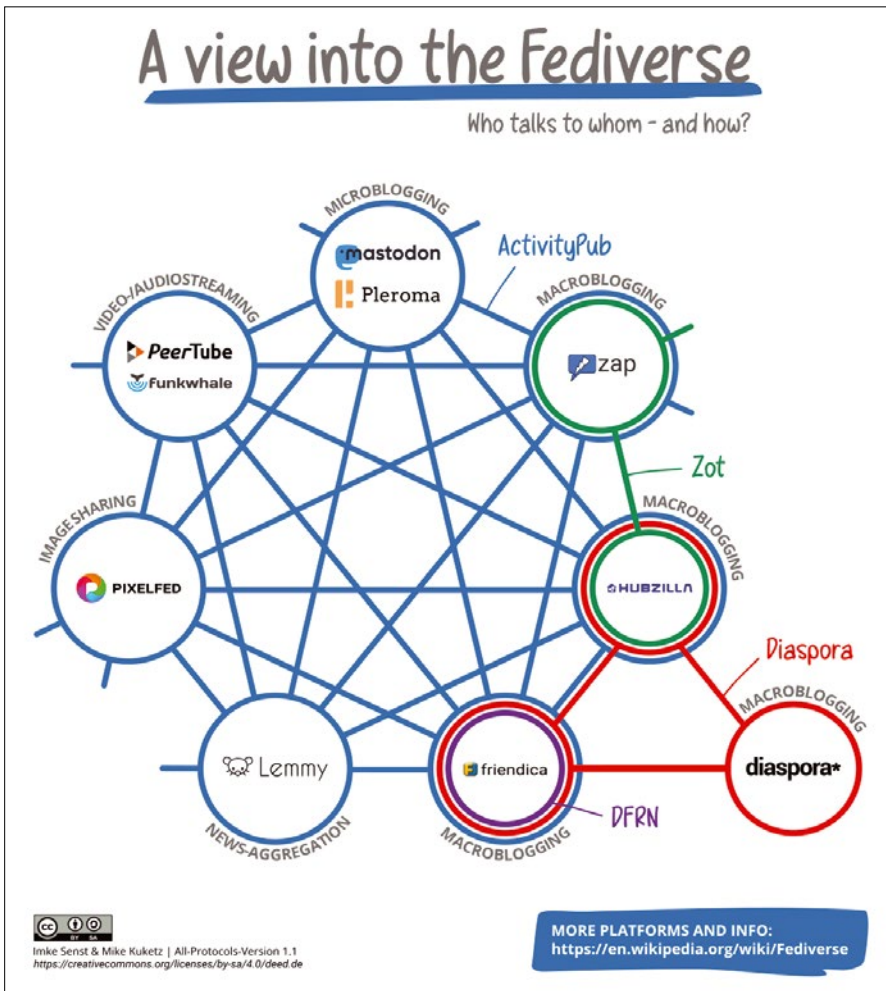

Figure 3: ActivityPub is the most common Fediverse protocol, but Diaspora and a few other services use alternatives.
Mike Kuketz, CC BY-SA 4.0, https://creativecommons.org/licenses/by-sa/4.0/

Info

[1] Eben Moglen comments regarding Facebook: *https://yro.slashdot.org/story/12/02/06/1828231/moglen-facebook-is-a-man-in-the-middle-attack*

[2] Fediverse: *https://fediverse.info/*

[3] Fediverse Services: *https://fediverse.info/*

[4] Nostr: *https://nostr.com/*

[5] Nostr on GitHub: *https://github.com/nostr-protocol/nostr*

Meet Me in the Stream

You don't need Facebook to keep up with your friends online. With diaspora*, you get a similar experience without the spying.

By Nate Drake

The diaspora* federated social network allows users to independently run *pods* (diaspora* servers) to interact with each other and the world in a "privacy first" way. Technically speaking, diaspora* is a macroblogging platform. You could say it's a Fediverse equivalent to Facebook. You can still create a space that serves as your online presence, post pictures, links, and personal reflections. But this time, you stay in control of your data.

Anyone can set up or join their own diaspora* server (called a *pod*) using free and open source software. Although pods are managed separately from each other, they can be federated, so someone who uses their account on the US-based *diasp.org*

can still interact with another user who's registered with the German-based *despora.de*.

Diaspora* (Figure 1) fills the strong need for a decentralized social network in a world where the established, monolithic social media services have abused their market dominance, playing fast and loose with users' privacy (see the box entitled "Facing Up to Facebook").

Decentralization

The creators of diaspora* drew their inspiration from a 2010 speech to the Internet Society by Columbia University Law Professor Eben Moglen, who described monolithic, centralized social networks as "spying for free"[5]. The dev team smashed their Kickstarter funding goals, and the first diaspora* pod was launched later that same year. The fact that diaspora* has been in development for almost 13 years gives it an edge over other decentralized social networks that are newer to the game and have had less time to work out the kinks.

Figure 1: Decentralization and freedom are two key benefits of diaspora*.

Author

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

Debutantes

While setting up your own server is the best way to control your personal information (see the box entitled "Setting Up a Server"), this approach takes time and resources. Fortunately there are a number of pods around the world run by eager volunteers. You don't even have to run to your nearest search engine to find them, as the Fediverse Observer website [9] maintains a list of servers running alternative social media instances, including diaspora* (Figure 2).

These lists make for sobering reading, as even the most popular diaspora* pods only host a few thousand users, compared to Facebook's billions. Getting started is usually just a question of going to a diaspora* pod's website and filling in your email address, name, and password. Some pods, such as *diasp.org*, require you to email to request an invitation link. Otherwise, setup is a breeze.

Facing Up to Facebook

A previous article in this magazine [1] discussed some of the more controversial uses Facebook (now Meta) found for their customer's data, such as retaining information after you close your account and using facial recognition to identify people in photos [2].

Meta doesn't seem to have learned its lesson. In November 2022, Ireland's data privacy regulator fined the company $277 million for alleged privacy breaches. This was the fourth fine of this kind Meta had received in Europe [3].

In December 2022, Meta also paid a record $725 million for its role in the long-running Cambridge Analytica scandal, where personal data about users was made available to third parties [4].

Since these fines related to cases that were several years old, Meta was quick to reassure users that they'd since updated their website, and users now have advanced privacy settings. However, it is important to keep in mind the words of Andrew Lewis, "If you are not paying for it, you're not the customer; you're the product being sold." If Meta didn't have a way of making money on your data, they wouldn't be in business.

Diaspora* uses hashtags (e.g., `#linux`) to help you record and search for your interests. When registering a new account, you'll be asked to enter a hashtagged topic in order to see relevant content on that topic in your *main stream*, which is a newsfeed by any other name.

When the main screen loads for the first time, a helpful wizard also guides you through diaspora*'s main features (Figure 3). As pods are run by volunteers who aren't profiting from selling targeted ads, new users are sometimes asked to make a small donation. This isn't compulsory.

Discretion

One huge advantage diaspora* still boasts over its bigger brother Facebook is that there's still no requirement to use your real name when registering an account (Figure 4), which will be welcome news to those who have been victims of stalking and online harassment.
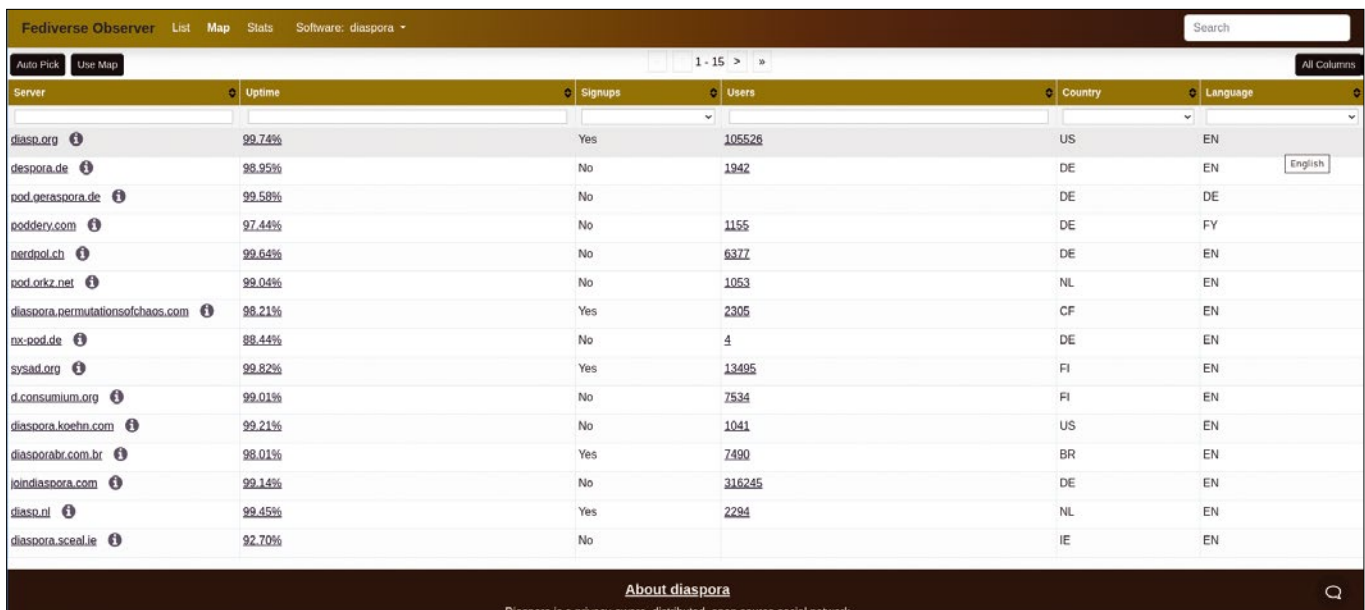
If this isn't enough incentive to make the switch from Meta, diaspora* also has much more clearly defined privacy settings, which are enabled by default. First and foremost is diaspora*'s use of *aspects*. Aspects are similar to the *circles* formerly used

Setting Up a Server

The lack of one monolithic server in diaspora* means that data is much less likely to be hacked or misused, particularly if you set up your own diaspora* pod using the foundation's source code, which is freely available via GitHub. [6]

Setting up a server should present no challenge at all to anyone who has previously run one. The diaspora* Foundation has very clear installation instructions, though at this time of writing the software won't run on Ubuntu 22.04 or Debian 11 [7].

The Network Admin FAQ points out the software is a standard Rails application, so it can run quite happily with Passenger or a reverse proxy configuration with Apache or NGINX. There's no requirement to use SSL, but pods that do must have a commercial certificate (i.e., they can't be self-signed) [8].

Figure 2: The Fediverse Observer at *https://fediverse.observer/* will help you find a diaspora* pod to join.
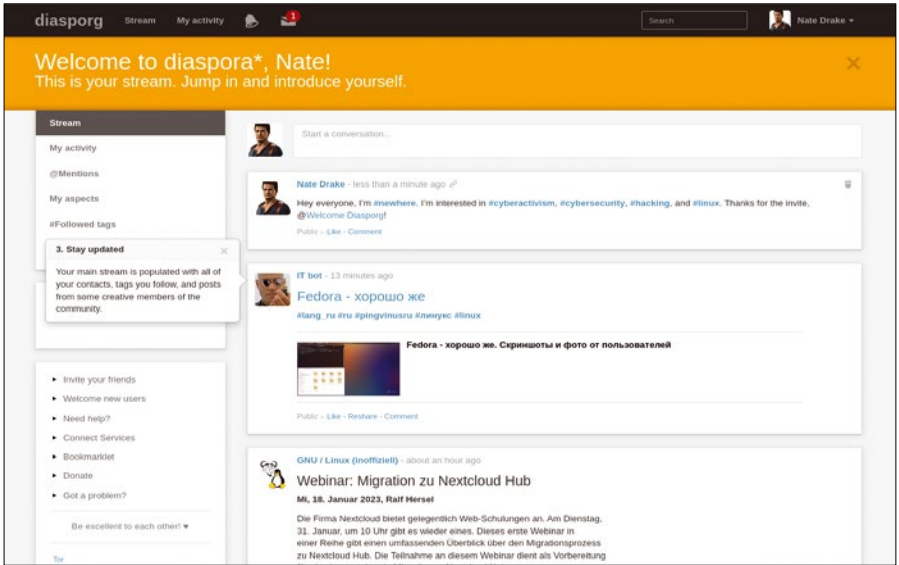
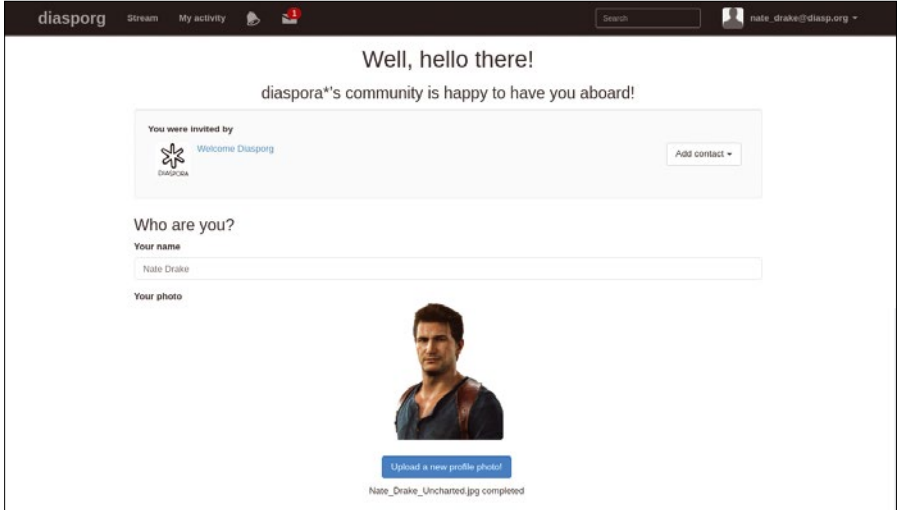**Figure 3: The diaspora* welcome wizard talks you through the sidebar and how to write your first post.**

Figure 4: Unlike Facebook, you can use any name you want when registering a diaspora* account.
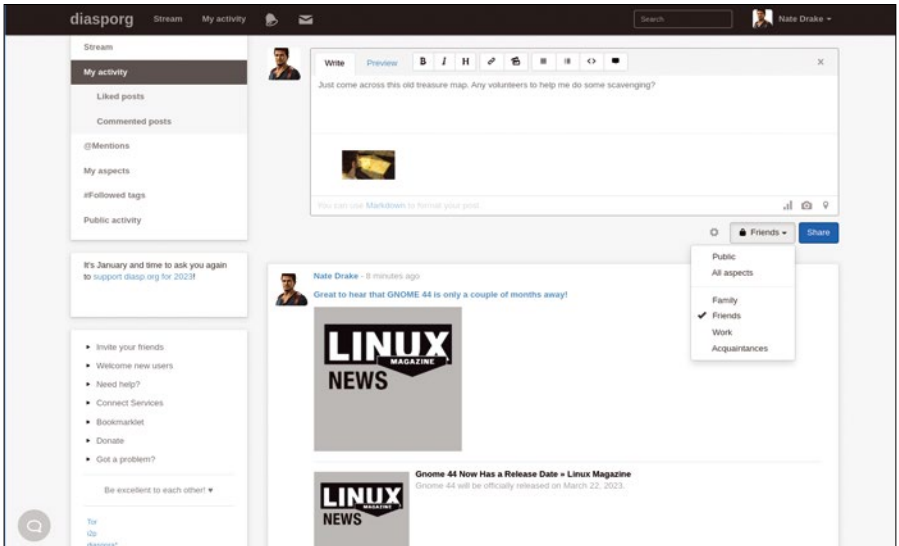

Figure 5: You can choose to share posts only with certain aspects like family, friends, work, or acquaintances.

by Google, as well as Facebook's *audiences*. When you create a post, you can choose who can view it.

The default setting is that posts are public but you can change this using the drop-down menu, so that only people in your "Family," "Friends," "Work," or "Acquaintances" can view what you post (Figure 5).

You can create new aspects, which is useful because diaspora* has no equivalent feature to Facebook's groups. You can also create an aspect with a specific group of contacts. Provided all of you add everyone else to that specific aspect, you can share posts only with each other. This is effectively the same as a private Facebook group.

Your contacts have no way of knowing to which of your aspects you've added them, unless you choose to tell them. You can also use the sidebar to see only posts from people in a particular aspect, such as your family [10].

If you create a diaspora* account, take a moment to review the rather minimalist privacy options by clicking on your name at the top right and then choosing *Settings*. Unlike Facebook, where you must manually enable most extra privacy features, diaspora* operates on an "opt-in" basis. For instance, by default you won't start sharing information with a contact just because they're sharing with you. (You can enable this of course.) From here, you can also change the default aspect for those who can see what you're posting from "Public" to something else.

Since our last review, diaspora* has also introduced a special web protocol. Any link to a diaspora* page on an external website that uses this protocol can be opened in the pod on which your diaspora* account is registered. This feature is still in the testing stage, and you must enable it manually.

The *Privacy* section of the account is even more spartan, though this may be because the developers have already thought of most ways you need to stay safe. During our tests, we saw that this section automatically enables stripping all metadata from images you upload, such as location, author, and camera model.

You can also enable two-factor authentication (2FA) from the Settings menu, as well as export all your user account data or photos. If a simple VPN isn't enough, some pods, such as *diasp.org*, are also

reachable via a Tor (`.onion`) hidden service and I2P (`.i2p`), though this is a matter for the pod admin to set up.

The Interface

Pod admins have the power to customize the diaspora* user interface, so the details can sometimes vary, but Figure 3 offers a good example of what you'll find when you log in to your diaspora* account. As mentioned previously, diaspora* uses the term *stream* to refer to the stream of posts you view when you log in. The posts can come from a number of different sources. Your stream aggregates content from your contacts and public posts made containing `#tags` you follow, as well as posts from others that mention you (diaspora*'s `@mention` feature). Diaspora* also supports an optional *Community Spotlight* feature that lets you see posts from members of the community that the pod admin recommends.

Figure 6: Although features are limited, you can post using Markdown, including inserting links and images.

Figure 7: Use the search bar at the top to find other users or posts tagged with your interests, like video games.

The *My activity* area shows posts that you have either liked or commented on. You can find it to the right of the *Stream* link in the top bar. The notifications icon (bell symbol) also lives here. The letter symbol is the conversations icon. In diaspora*, conversations are private communications between two or more people – similar to chat. At the far right of the top bar is a menu dedicated to your personal account, with options for contacts, settings, and a personal profile. As you can see in Figure 3, diaspora* also has a menu on the left side of the page that lets you manage aspects, mentions, and `#tags`.

Direction

Diaspora* offers many more options for creating posts than on most social media sites. You can add images and links but also use Markdown to format your text, add lists, and even poll other users.

Creating a post in diaspora* is called *publishing*, and the text window where you enter the post is call the *publisher*. The interface for entering the text appears quite sparse (see the publisher at the top of the window in Figure 6), but, as the diaspora* documentation states, "beneath its minimalist appearance lies a treasure trove of features, which you can trigger by clicking inside the box." Enter the text and apply the desired formatting options, and then press the *Share* button to share the post.

We were disappointed to see that, after all this time, it's still not possible to edit existing posts to correct typos or change their visibility. The developers agree that this would be very helpful but are focusing their efforts on federating servers for the time being. They are, however, offering a bounty to any developer who implements this feature on their pod [11].

You can use the search bar to find other users both on your pod and others (by adding the @ symbol and the name of their pod, for example, a *dias.org* pod user could search `macumazan@nerdpol.ch`). The search bar (Figure 7) also lets you find subjects that match your interests by finding posts with matching hashtags.

You can also comment and reshare other user's posts to your own *My Activity* feed, but as with your own posts, bear in mind these can't be edited.

Sharing

As with other social networks, users in diaspora* make decisions about who they wish to follow and who they wish to have following them. In diaspora*, following and being followed are theoretically independent (you can follow someone who doesn't follow you), but mutual sharing
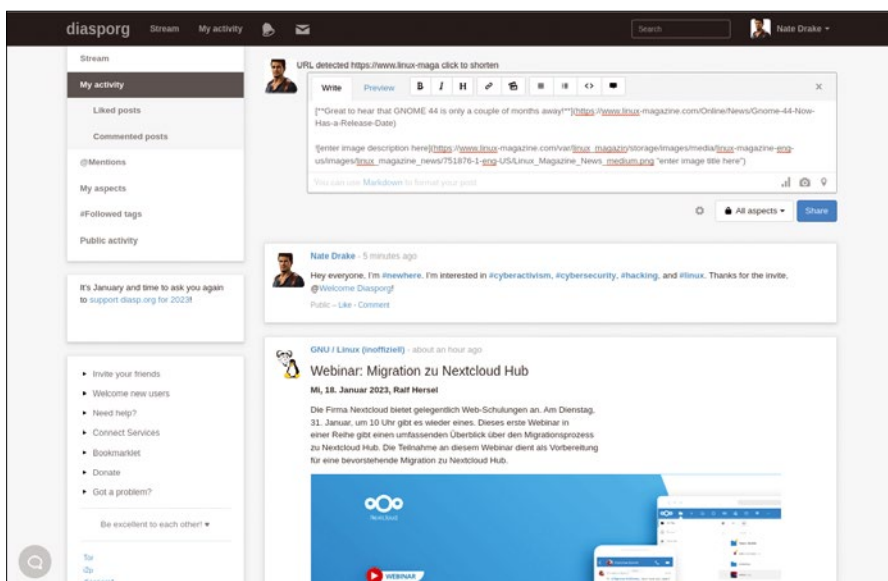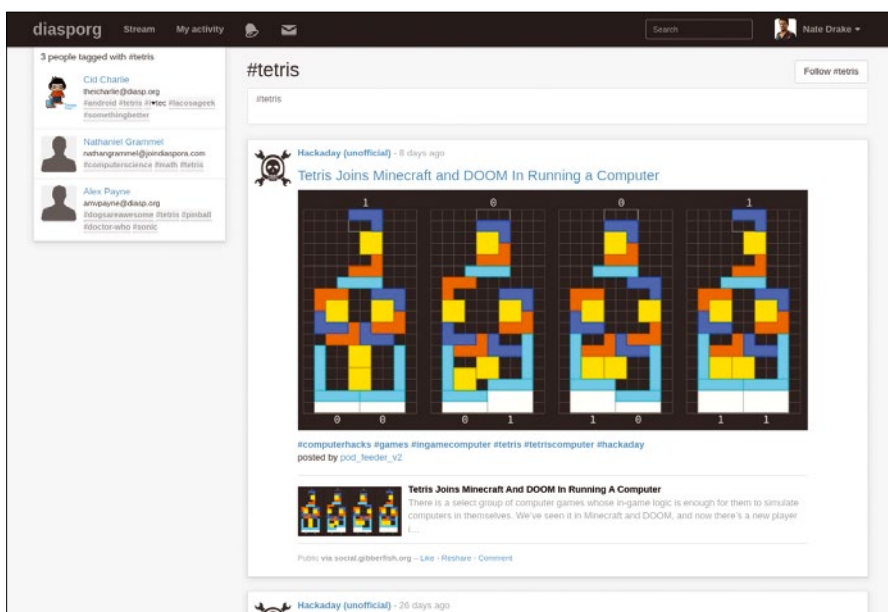
is a more common approach, especially among groups of friends. The versatility of the aspects feature means that sharing relationships are not always parallel. For instance, you might put the user in the group of your "close friends," but the user might put you in the group of "work friends." This means each of you might be viewing different content streams [12].

If your search for a name or diaspora* ID displays a user you'd like to connect with, just add them as a contact, and then add that contact to an aspect. You can also add a user to an aspect using their profile. When a user's name turns up in diaspora*, it is often a link to the user's profile. Click on the name to see the user's profile. Click the button in the corner of the profile page to add the user to an aspect.

You can also use email to invite other users. Click the links in any search results page to reach out to users by email. You also can click the *Invite your friends* link in the menu on the side of the *Stream* page (Figure 6).

Drawbacks

In 2014, diaspora* managed to garner the wrong sort of press attention when some ISIS members managed to set up their diaspora* pod in order to spread extremist material. The developers admitted there was little they could do to stop this given the network's decentralized nature but encouraged "podmins" (pod admin) to take down public posts of such material, most of whom did [13].

Naturally this works both ways: A group of human rights workers operating in a rogue state could also set up their own private pod to share news from western agencies and other censored information with citizens. Still, using diaspora* in a safe way requires a trustworthy and competent podmin.

Instant messaging using Prosody as an XMPP server is still in the development phase, and it's still a matter for each podmin to enable it or not [14]. This means it's not currently possible to chat in real time with other diaspora* users.

Unlike Facebook, there's no way to tag other users in posts, comment on individual photos, or arrange photos into albums. The diaspora* team planned to address this by using the `#tag` feature to group photos, but since my previous article in 2017, it seems there's been little work on this [15].

It's unlikely there will be much uptake in diaspora* usage while these features we take for granted on monolithic social networks aren't implemented. Still, this chicken and egg argument has its limitations: the more people who are involved, the more work can be done on introducing new features.

Some contributors have suggested federating diaspora* with other "open" social networks. The software is actually already compatible with any other network that uses its Federation protocol [16], such as Friendica and Hubzilla. Still, a universal solution that provides seamless compatibility with the ActivityPub protocol used with Mastodon and other Fediverse services seems a long way away.

Diaspora*'s desire for federation also means the pods aren't designed to be run in isolation, and the developers explicitly explain that's not what the network is for [17]. This is a shame, because one great use case for diaspora* would be to allow an offline social network in controlled environments like schools and prisons.

The Faustian bargain we make with social media giants like Meta means that they have the resources to hire crack developers to develop new features and expand their network, leaving diaspora* and its team of volunteers far behind. Still, if you manage the server yourself or can trust your podmin, diaspora* offers much greater control over the personal data you hold and share, even if its features are still rather spartan. ∎∎∎

Info

[1] "Social Networking the FOSS Way with Diaspora" by Nate Drake, *Linux Magazine*, issue 194, January 2017: *https://www.linux-magazine.com/Issues/2017/194/Diaspora*

[2] "7 Controversial Ways Facebook Has Used Your Data" by Victor Luckerson, *Time*, February 4, 2014: *https://time.com/4695/7-controversial-ways-facebook-has-used-your-vdata/*

[3] "Irish regulator fines Facebook 265 million euros over privacy breach," CNBC, November 28, 2022: *https://www.cnbc.com/2022/11/28/irish-regulator-fines-facebook-265-million-euros-over-privacy-breach.html*

[4] "Meta settles Cambridge Analytica scandal case for $725m" by Shiona McCallum, BBC, December 23, 2022: *https://www.bbc.com/news/technology-64075067*

[5] ISOC-NY Event: Eben Moglen "Freedom in the Cloud," February 5, 2010: *https://isoc-ny.org/1338*

[6] diaspora*: *https://wiki.diasporafoundation.org/FAQ_for_pod_maintainers*

[7] Installation: *https://wiki.diasporafoundation.org/Installation*

[8] General system requirements: *https://wiki.diasporafoundation.org/FAQ_for_pod_maintainers#What_are_the_general_system_requirements.3F*

[9] Fediverse Observer: *https://fediverse.observer/*

[10] FAQ for users: *https://wiki.diasporafoundation.org/FAQ_for_users#What_is_an_aspect.3F*

[11] Current and future development: *https://wiki.diasporafoundation.org/Current_and_future_development#Editing_posts*

[12] The diaspora* interface: *https://diasporafoundation.org/getting_started/interface*

[13] "Diaspora social network cannot stop IS posts" by Dave Lee, BBC, August 2, 2014: *https://www.bbc.com/news/technology-28882042*

[14] Implement realtime chat, possibly using XMPP: *https://discourse.diasporafoundation.org/t/implement-realtime-chat-possibly-using-xmpp/189/353*

[15] Photo organization: *https://discourse.diasporafoundation.org/t/organising-photos/253/26*

[16] Federation: *https://wiki.diasporafoundation.org/Federation*

[17] Isolation: *https://wiki.diasporafoundation.org/FAQ_for_pod_maintainers#Can_I_make_my_pod_private.2Fisolated.2Fnot_communicate_with_other_pods.3F*

Want to work from anywhere?

Find remote jobs now!

OpenSource
JOB HUB

opensourcejobhub.com/jobs

Elephant in the Room

Get started with free microblogging, interact with others who share you interests, and even help expand the Fediverse with your own Mastodon instance. *By Paul Brown*

Recent affairs at Twitter have led to resounding and sudden success for the free, open, and federated Mastodon [1] social media platform. But despite the apparent benefit of Twitter's mismanagement, Mastodon's success was bound to happen sometime. As Cory Doctorow explains in one of his many eye-opening essays [2], proprietary, centralized, and for-profit social media platforms always kill themselves. Mastodon and other open platforms will fill in the vacuum they leave in their wake.

Mastodon Is Not Twitter

Thinking of Mastodon as a stand-in for Twitter will leave newcomers confused. Sure, on the surface, Mastodon is a microblogging social media platform, and you could use it a such. But, as Max Leibman points out, assuming that Mastodon is just like Twitter is the first of the two fundamental errors people make.

Mastodon taps into and inherits all the advantages of the Fediverse. It is not only distributed, but massively distributed. This means no controlling entity can mess with your content or push sponsored posts onto your feed. Although a company can buy up individual servers, it will be hard for Mastodon, as an ever-expanding network of interconnected, but independent instances, to be bought out.

Mastodon is also a door into the Fediverse. Built on *Activity-Pub* [3], a W3C recommended standard for social media platforms, it interconnects with other Fediverse services in ways that are both surprising and surprisingly useful.

How to Mastodon

You can experience Mastodon without even signing up. Visit, for example, *Linux Magazine*'s feed [4] and you can scroll through the news we publish there without interference – and without the annoying pop-ups from other platforms that order you to register to continue browsing.

Click on *Local* on the right (see Figure 1) and you will see the posts of all the accounts hosted on that particular "instance," in this case, *fosstodon.org* (more on instances in a minute). Click on *Federated* and you will see all the messages being posted to all the instances that have federated with *fosstodon.org*.

Ready to take the dive? To sign up, your first step is to choose an instance. Each instance is a Mastodon server that is part of the network of Mastodon servers. Each instance is owned by a different entity, which could be a person, a group of friends, a community, or an organization. The owner decides if the instance will be based around a topic, what the rules will be, and what the registration steps will be. Not all instances have open registration. Some do, but for some you need an invitation. Still other Mastodon instances are completely closed to new users.

You might think it is best to go with an instance that has lots of users, because it is less likely to disappear (yes, instances can and do disappear from time to time – the same as with any Internet service), but that is not always the best choice. Bigger instances have more traffic and are often slower. They are also more difficult to police, and, hence, more likely to have spammers posting bad things. A mid-sized instance with, say, 1,000 users or fewer, is often a better bet.

Choose an instance that has rules that sit well with you regarding content, harassment, veracity, and hate speech. Just in case, before signing up with an instance, browse the content posted by other users – as with everything, there are disreputable *tooters* (Mastadon posters).
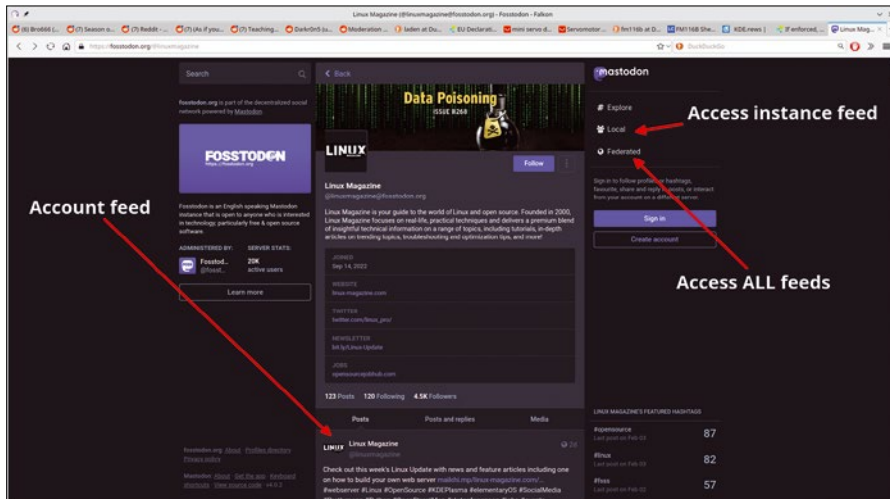
Figure 1: *Linux Magazine's* **Mastodon account and feed.**

If you have a specialized job or hobby, you might want to check if there is an instance specializing in your interests. You will also find instances tied to specific languages and locations. Table 1 shows some suggestions. Check the Mastodon website for a more complete list [5].

Remember, all instances can talk with each other (except those that are blocked) and you can follow and interact with accounts that are on other instances.

Once you have registered and logged in, setting up the basics of your account is pretty straightforward: Click on *Preferences* (bottom option in the right-hand column), and under *Profile | Appearance*, you can set up your display name, load an avatar, configure a header image, enter a bio, and set the visibility of your account.

I also like to go into *Preferences | Appearance* and tick the *Enable advanced web interface* checkbox, then click the *SAVE CHANGES* button at the top of the page. That gives you a multi-column view (Figure 2), as opposed to the default single column.

It Follows

The whole point of all this is to be able to follow, be followed, and to read and share messages, or *toots,* as they are known in Mastodon parlance. But the first thing you will to want to do is decide what you want to see.

Notice the last column on the right of your multi-column view. That is the *Getting Started* column. Click on *#Explore* and it will open a new column that gives you a list of toots from all over the Fediverse that are getting a lot of *boosts* (shares) at that moment in time, the most used hashtags, news, or a list of accounts you might be interested in. The algorithm that selects the suggested accounts is intentionally unsophisticated. It recommends mainly "friends of friends," that is, people that your followers and people you follow follow.

There is no data analysis in Mastodon and no profiling. The rules for recommendations are extremely basic. There is

a good reason for this: The creators of Mastodon want to encourage human-to-human interaction, as opposed to human-to-algorithm. Too often, in less scrupulous social media platforms, users ended up writing posts to gain the favor of the machine. Also, as Mr. Musk, in a moment of rare candor, points out in Figure 3, there are other drawbacks to behavior-based recommendations.

When it comes to building relationships, the onus is on you, but Mastodon's "recommendation" system is a good place to start for setting up your network of contacts.

Maybe more interesting are the local and federated timelines. Click on the *Getting started* hamburger menu (on the left, just above your profile picture) to bring back the original column on the right, and click *Local Timeline*. This option will display a column with all toots sent from accounts on your instance. If you are on a busy server, there will be a lot going on. If you are not, well, there won't.

In the *Local Timeline* column, on the left, you will notice a small *Settings* button. Click it and then click *+Pin* to attach it to your columns. Once it is pinned, a new *Getting started*

Table 1: Specialized Instances

| Instance	Topic
astrodon.social	Astronomy
mastodon.art	Art
mstdn.games	Gaming
mastodon.gamedev.place	Game development
fosstodon.org	Free/Open Source Software
disabled.social	Accessibility
floss.social	Free/Open Source Software
mathstodon.xyz	Mathematics
federated.press	Press/Journalism
mapstodon.space	Cartography/Maps
veganism.social	Veganism
metalverse.social	Metal music
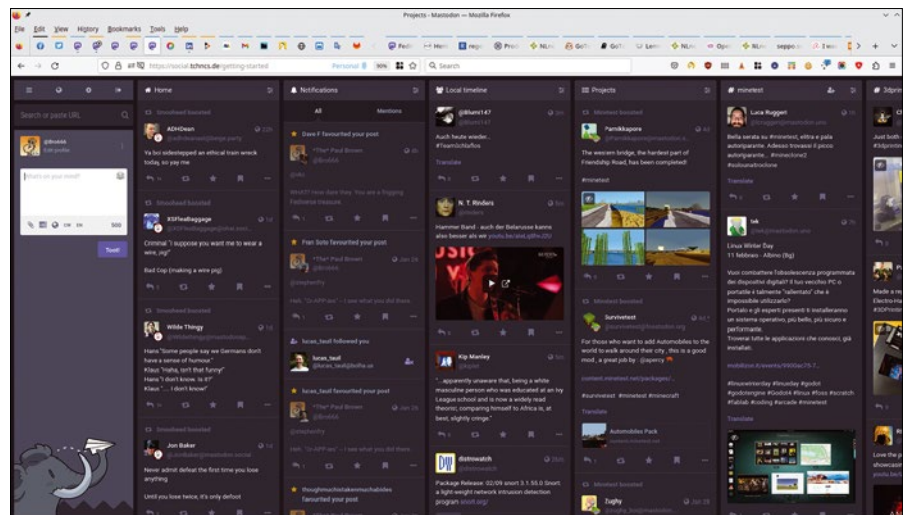
Figure 2: The Advanced web interface gives you a multi-column view.

Figure 3: A moment of clarity from Mr. Musk.

column appears. You can do the same with the *Federated time-line* option and pin that to your interface too. That column will show all toots being sent to all instances in the Fediverse that your instance can see, so the *Federated timeline* is much busier than the *Local timeline*.

To follow someone, click on their username at the top of a post, and their feed will open in the column on the right. Browse their feed to be sure you are really interested in reading what they have to say. When you are sure, click the *Follow* button shown in their profile.

There is one account you will want to follow from the beginning: Copy *@FediFollows@social.growyourown.services* into Mastodon's search box and click on the *Follow* icon. This account posts regular updates with curated lists of interesting people.

In these days of growth for Mastodon, with lots of new users landing on the platform, following someone is a good way of gaining a follower: There is an informal agreement among new users that if someone follows you, it is polite to follow back.

Of course, you are more likely to be followed if you have something to show in your timeline. Which brings us to …

Regarding Posts

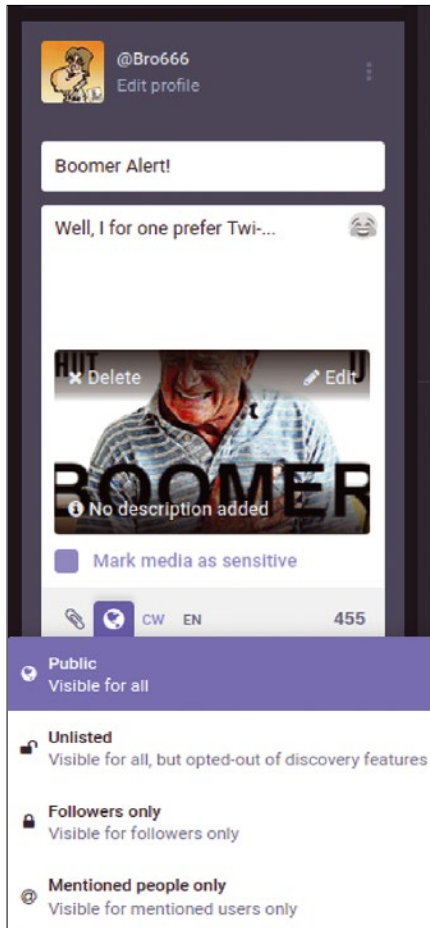You will have noticed by now the text box in the
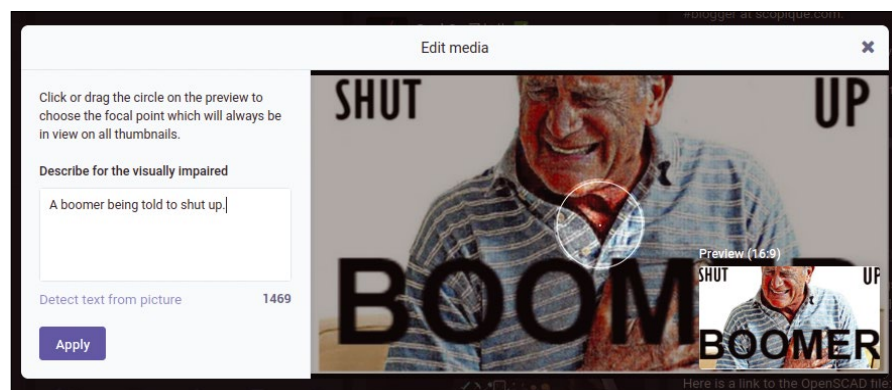

Figure 4: Parts of the tooting form.

upper left of the web interface. This is where you compose your toots. But before you shoot off your first pearl of wisdom into the Fediverse, notice how Mastodon differs (Figure 4) from other platforms.

First, you have 500 characters for your message, which, well, is nice. Notice along the bottom of the main text area, and from left to right, is a toolbar with five buttons. The first button shows the universal visual symbol for attachments: a paperclip. The paperclip is, of course, for attaching files to your toots. You can also just drag and drop a file onto the web client. Mastodon allows up to four images or one audio/video file per toot.

Note that accessibility is very important for the Mastodon community, so you are highly encouraged to add *alt* (alternative) text to the images you attach. Indeed, many servers surround images that do not do this with a red and yellow warning border. Others are more aggressive and directly forbid posting media without descriptions or have a bot remove them. So, when you attach your media, click on the *No description added* label in the attachment itself and a new window will open (Figure 5). In this window you will be able to add text of up to 1,500 characters.

The next button is for polls and does what you would expect.

The button with a globe is for the privacy level of the toot. The privacy levels are self-explanatory, but bear in mind that the *Mentioned people only* option, that is, a direct message to other users, is not very private. Messages in Mastodon are not encrypted, so do not use it to send private or sensitive information.

And another etiquette note here: When you want to create a thread (a series of interconnected toots), the done thing is to make the first one *Public*, and the follow-up toots *Unlisted*. That means that anybody who wants to read the thread can just click on the top, public toot and see all the other threaded toots, but people who are not interested will only see the first toot and you won't clutter up their feed. This also applies to when you answer a toot: The polite thing to do is send your reply as unlisted.

The *CW* button is also something unique to Mastodon. *CW* stands for "content warning" and splits the text field into two. If you are posting content that could show something disagreeable or distressing to the rest of the world, you indicate the fact here, in the content warning text box. The rest of the toot will then be hidden and images and videos blurred (Figure 6). A user who wants to read the rest will be given a button to press.
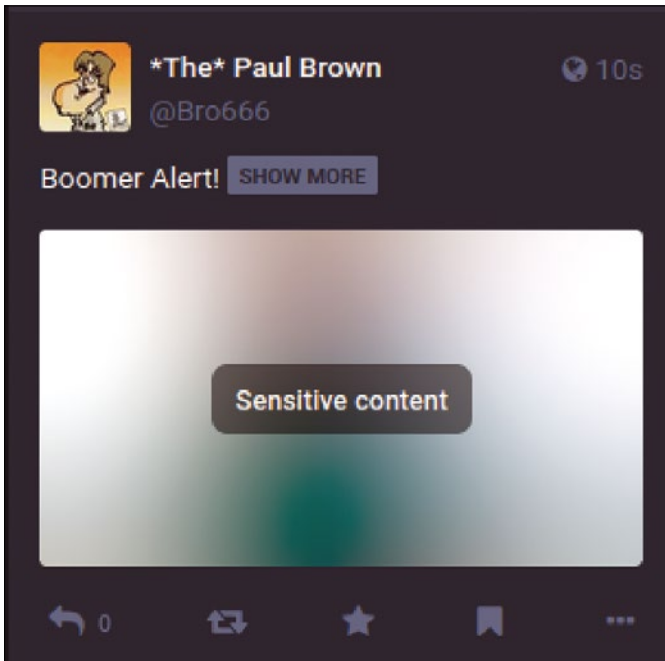

Figure 5: Adding alt text to an image.

Figure 6: A toot with a content warning.

A less-than-obvious real-life example is that a lot of people on Mastodon have grown a bit tired of all the mentions of Twitter, so a warning with the words "Birdsite Rant" will tell those people that they should skip this one.

Again, it is a Mastodon thing to use this feature generously to avoid spoiling the day (or wasting the time) of fellow users.

Verifying

Your tooting prowess will eventually build up your following. But with fame also comes envy! And with envy comes identity theft, or at least someone trying to pass off as you. Notice that, if you are using @JaneDoe on one server, for example, someone can create the account @JaneDoe on another server and try and mislead your followers [6].

However, Mastodon implements an ingenious workaround that lets you verify your identity to your followers. It has nothing to do with colored ticks, because you use your website.

Visit *Preferences* (gear icon at the top of the web client screen) and then *Profile | Appearance*. You will see a section labeled *Profile metadata*. You can put anything in there, but most users use it to provide information such as links to their blogs, websites, Patreons, and so on. Use one of the rows to write in the URL of the front page of your website or blog (Figure 7).

Under *Verification*, to the right of *Profile metadata*, you will see a string of HTML. Use the *Copy* button to copy it to the clipboard, and open your web page for editing. Insert the string in the `<header>` ... `</header>` section of your page.

(For WordPress, you can try a plugin such as WPCode [7]. WPCode allows you to insert snippets into different parts of your blog. Once installed, it adds a new entry to the sidebar: *Code Snippets*. Click on that entry and choose the *Header & Footer* submenu.)

Save your changes on your website, and, after a few minutes, the links on your profile will show up in green with a green tick (Figure 8), verifying you as the owner of both the linked site and the Mastodon account.
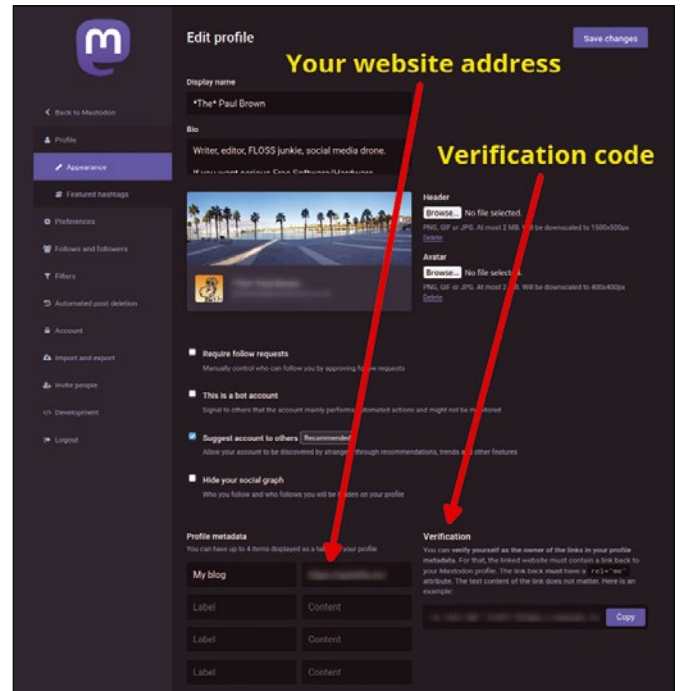

Figure 7: Add your website or blog to your profile to verify the authenticity of your account.

Interoperability

One of the great aspects of Mastodon is how well it plays nice with the rest of the Fediverse. Take for example Pixelfed [8], a Fediverse service for sharing photos, somewhat akin to Instagram. Look at the name of the accounts and you will notice that they follow the same format as in Mastodon: @ [name]@[instance].

This means you can follow a Pixelfed account from Mastodon (and vice versa) (Figure 9), and every time your favorite photographer posts an image, it will show up in your Mastodon feed (see the article on Pixelfed elsewhere in this issue). Posts you boost and like from Mastodon will be boosted and liked in Pixelfed too. If you send a reply to a post, it will show up as a comment in Pixelfed.

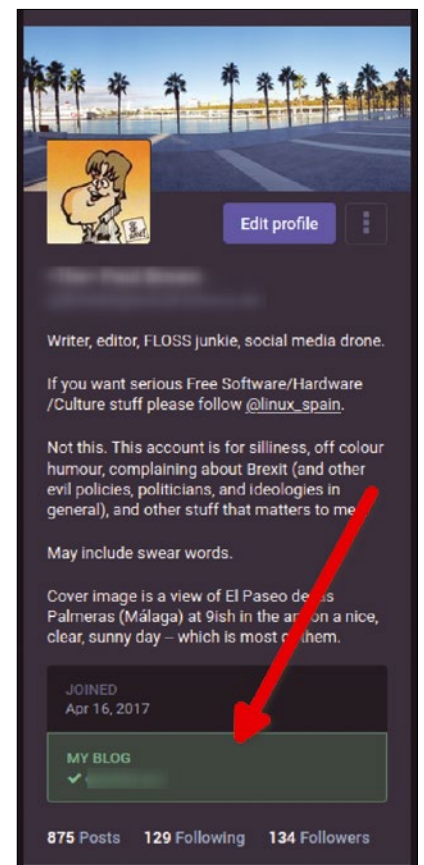Likewise, you can follow a Mastodon account from Pixelfed, and every time

Figure 8: A verified account.

that account posts a toot with an image, it will show up in Pixelfed.

It is the same for PeerTube [9], the Fediverse's service for sharing videos (also covered elsewhere in this issue): Follow a Peer-Tube account from Mastodon, and new videos will appear in your feed, and you can watch, boost, like, and comment on them without ever leaving Mastodon.

Mastodon on Mobile

The mobile space supports dozens of clients for Mastodon (and other Fediverse services) [10]. The one you choose is a question of taste. I recommend choosing one that supports *scheduling*.

The ability to set a toot to be posted in the future, at a given time and on a given date, is baked into Mastodon's architecture and is thus available on all instances by default. You don't need a special scheduling service as you would on other platforms, but you do need a client that can leverage the feature.

My favorite client for Android is Fedilab [11].

Fedilab is free, it is powerful, and it easily maintains multiple accounts, and it does scheduling well (Figure 10).

Migrating

There may come a time when you need to move to another instance. It could be that the instance you are on is shutting down, or has gotten too busy, or it has been overtaken by trolls, or the new one simply fits your preferences better. Whatever the reason, Mastodon makes it relatively easy to take your configuration, the list of people you follow, and even your followers with you.

Before you migrate, bear in mind that Mastodon does not allow taking your old posts with you to your new instance. It would be theoretically possible to copy the contents from your toots from one database to another if you had system administrator access to both servers where the instances are hosted. But your posts are time-stamped, and not only on your instance, but also on every instance where people interacted with them. The timestamps of imported toots would not match the originals, and that would make an inconsistent mess in the federation. Besides, apparently all of your "new" old toots would be posted a second time all at once, flooding followers' timelines and in general wreaking havoc. So one thing you must not do is delete the old account, because that is where your original toots will and should stay.

There is a second good reason to not remove your old account entirely: Mastodon can automatically forward followers who visit your old account to the new account. If there is no old account, Mastodon will not know who to forward nor where to forward them to.

Let's say you are migrating your "JaneDoe" account from the *mastodon.social* instance to *myinstance.org*.

You can start by downloading your archive. To do that, in *Preferences* visit *Import and export*, choose *Data export*, and click *Request your archive* (Figure 11). If you have posted a lot, this may take some time, because it compiles all your toots and all their attachments into one compressed archive. The archive is not strictly necessary, because, as I said, you will not be able to import your toots into your new account. But it is a good backup, and the archive also contains content such as your profile and header image, the text you used in your bio, and so on. Having all these things in one place is just convenient when setting up your new account and making it look and read like your old one.

While you are at it, and in the same screen, download the files in the *Data export* table:
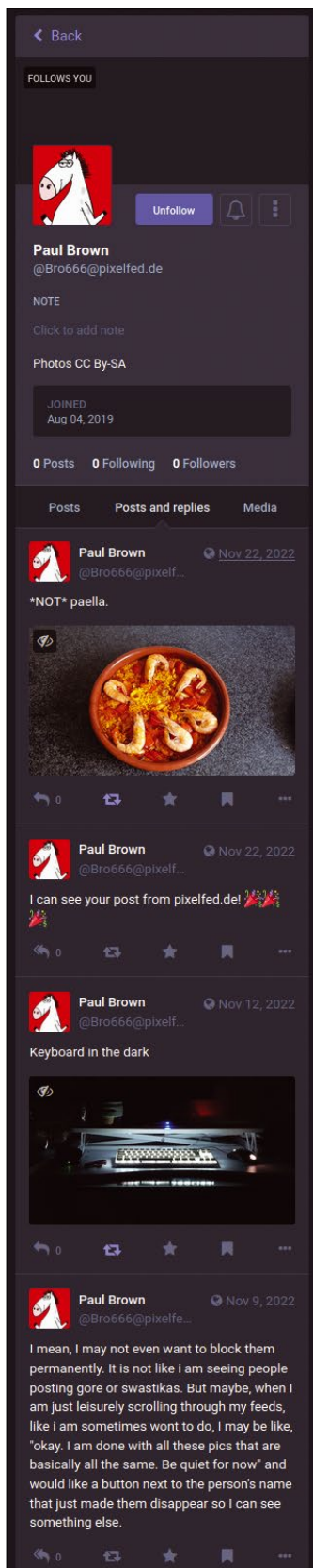

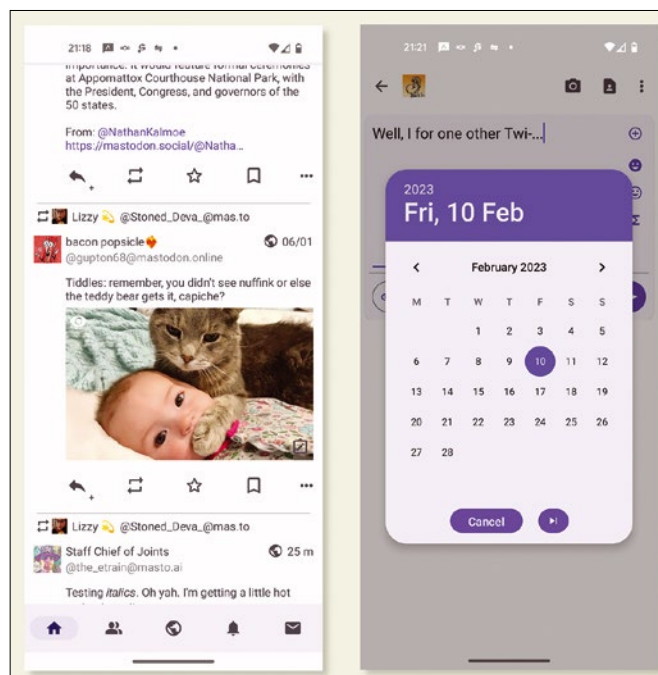**Figure 9: A Pixelfed feed viewed from Mastodon.**

Figure 10: Fedilab supports multiple accounts and toot scheduling.

- *Follows* contains the list of people you follow.
- *Lists* contains a list of the lists for grouping accounts you made.
- *You block* contains a list of accounts you blocked.
- *You mute* contains a list of accounts you muted.
- *Domain blocks* contains a list of domains you blocked.
- And *Bookmarks* contains a list of references to toots you have bookmarked.

You will be able to import these files into your new account.

In fact, you can do that now: Register your new account, and set up your profile from the contents of your archive. Then go to *Preferences | Import and export | Import* and use the CSV files you downloaded from your old account to update your new account.

Do not leave your new account. Visit *Preferences | Account Settings* and click on *Create an account alias* under the section called *Moving from a different account* located down towards the bottom of the page.

In the text box, write in the user name (with the instance name) of the old account, in this case, *JaneDoe@mastodon.social*, and click *CREATE ALIAS*.

Go back to your old account and visit *Preferences | Account | Account settings*. In the *Move to a different account* section, click on *Configure it here*.

Fill in the *Handle of the new account* field with the user name and instance of your new account, in this case, *Jane-Doe@myinstance.org*, and type in the password of your old account in the *Current password* text box to confirm the move.

Click *MOVE FOLLOWERS*.

Two things to note: The first is that transferring followers takes time, sometimes several days. Moving happens as your followers log in to Mastodon, browse your stuff, or see your new posts. That is when they are transferred. It might be a good idea to warn users beforehand, so they don't think something nefarious is going on.

The other thing to notice is that, from the moment you press *MOVE FOLLOWERS*, posting from the old account will not be possible. That said, unless you delete your account, all your posts will still be available, so nobody's links will break.

Personal Instance

The Fediverse becomes stronger if more people support it with their own instances. More instances allows for better distribution of the load of sending posts. Instances with many accounts can get bogged down with traffic, so having the same number of accounts spread over a larger number of federated instances actually makes the whole system faster. Bigger instances also become more vulnerable to abuse and spam. Specialized instances encourage diversity. And, as the number of Mastodon users grow, more instances make it more difficult to imagine a centralized, controlling hub gaining influence over the platform.

The path to creating an instance begins with securing your server. I will not dwell on the general security issues, because many of these same issues apply

to all Internet-facing servers. A couple of hints, though: Look into non-password access for accessing your machine over SSH, install and configure a decent firewall, and deploy something like Fail2Ban [12] to ward off brute-force attacks.

Once your machine is safe, you will need the NGINX web server to run your instance on and PostgreSQL for Mastodon's database, so install both:

```
apt install nginx postgresql postgresql-contrib
```

Enable and start NGINX with

```
systemctl start nginx; sudo systemctl enable nginx
```

It is also useful to install cURL and Wget, as you will need to download items during the installation. Also install gnuGP and CA certificates for keys and certificates:

```
apt install -y curl wget gnupg ca-certificates
```

Another necessary component is Node.js, which manages the live streaming of toots to different feeds:

```
curl -sL https://deb.nodesource.com/setup_16.x | bash - apt-get ⤷
install -y nodejs
```

To satisfy some of Node.js's dependencies, also install Yarn, a package/project manager:

```
corepack enable
yarn set version classic
```

and then install several other packages to satisfy Mastodon's specific dependencies (see Listing 1).

Setting Up Ruby

Mastodon is written in Ruby, and in the next steps, you will create a *mastodon* user and install the tools to build and set up a Ruby environment. In this environment, you will install and run Mastodon.
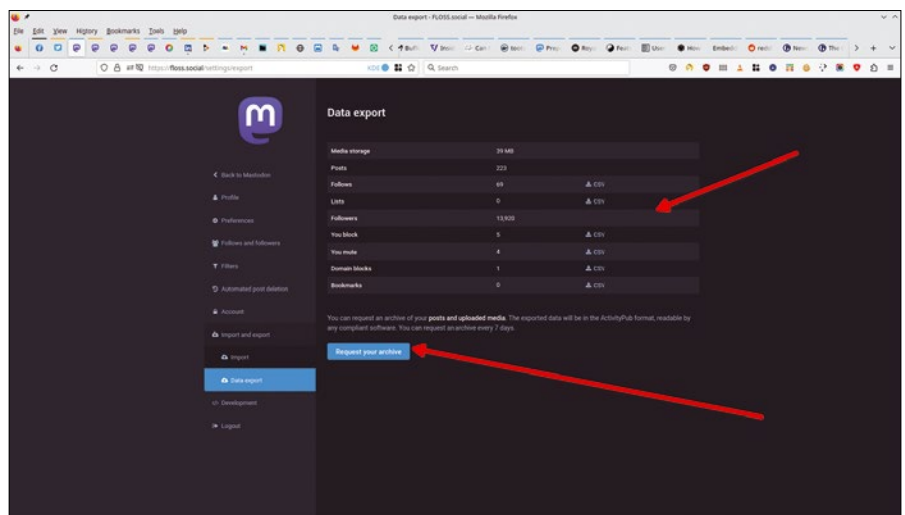

Figure 11: When migrating, start by downloading your archive from your old account.

Listing 1: Mastodon Dependencies

```
apt install -y imagemagick ffmpeg libpq-dev libxml2-dev libxslt1-dev file git-core g++ \
libprotobuf-dev protobuf-compiler pkg-config gcc autoconf bison build-essential libssl-dev \
libyaml-dev libreadline6-dev zlib1g-dev libncurses5-dev libffi-dev libgdbm-dev redis-server \
redis-tools certbot python3-certbot-nginx libidn11-dev libicu-dev libjemalloc-dev
```

Make sure you create the user without password access, so you can only access it via root:

```
adduser --disabled-login mastodon
```

Log into the new *mastodon* user:

```
su - mastodon
```

Install the tools to set up the Ruby environment (Listing 2).

Finally, install the correct version of Ruby:

```
RUBY_CONFIGURE_OPTS=--with-jemalloc rbenv install 3.0.4
rbenv global 3.0.4
```

You will also need to install Bundler, a package manager for Ruby gems:

```
gem install bundler --no-document
```

Configuring PostgreSQL

Exit the *mastodon* user to go back to root and configure your database manager. Tweak the file located at `/etc/postgresql/VERSION NUMBER/main/postgresql.conf` to best adapt it to your hardware. Then restart the server with:

```
systemctl restart postgresql
```

Next you need to create a database user for your instance, so log into PostgreSQL:

```
sudo -u postgres psql
```

And run the following at the `psql` prompt:

```
CREATE USER mastodon CREATEDB;
```

This creates both a *mastodon* user and a *mastodon* database associated with the user.

Listing 2: Tools for Ruby Environment

```
git clone https://github.com/rbenv/rbenv.git ~/.rbenv
cd ~/.rbenv && src/configure && make -C src
echo 'export PATH="$HOME/.rbenv/bin:$PATH"' >> ~/.bashrc
echo 'eval "$(rbenv init -)"' >> ~/.bashrc
exec bash
git clone https://github.com/rbenv/ruby-build.git ~/.rbenv/plugins/ruby-build
```

Listing 3: Getting the Mastodon Code

```
git clone https://github.com/mastodon/mastodon.git live && cd live
git checkout $(git tag -l | grep -v 'rc[0-9]*$' | sort -V | tail -n 1)
```

To exit `psql` type `\q` and press Enter.

Setting Up Mastodon

To download and install the actual Mastodon code, you again switch to the *mastodon* user:

```
su - mastodon
```

and clone and check out the latest version of the Mastodon server code (Listing 3).

Install some more dependencies for Ruby and JavaScript (Listing 4).

And you can now run Mastodon's built-in configuration wizard:

```
RAILS_ENV=production bundle exec rake mastodon:setup
```

The configuration wizard will ask you whether you want to run your instance for only one person (supposedly you) or if you will be accepting other users. It will also ask you to provide details about an email address for the maintainers. With this information, the wizard will set up an admin user and give you a one-time, randomly generated password so you can log in.

Note that all the info created by the wizard is stored in a hidden file called `.env.production` in the top Mastodon directory. You can change any of the values by editing that file.

NGINX Configuration

Mastodon comes with a sample configuration file that you can copy over for NGINX to use. So set up your `sites` directory for NGINX:

```
mkdir /etc/nginx/sites-available
```

and copy it over:

```
cp /home/mastodon/live/dist/nginx.conf ↵
  /etc/nginx/sites-available/mastodon
```

You will also have to enable it by creating a soft link in the `nginx/sites-enabled` directory:

```
mkdir /etc/nginx/sites-enabled
ln -s /etc/nginx/sites-available/mastodon ↵
/etc/nginx/sites-enabled/mastodon
```

Edit the file to adapt it to your instance (see Listing 5), and add the following line:

```
include /etc/nginx/sites-enabled/*;
```

in the `/etc/nginx/nginx.conf` file within the `http` section.

Listing 4: Ruby and JS Dependencies

```
bundle config deployment 'true'
bundle config without 'development test'
bundle install -j$(getconf _NPROCESSORS_ONLN)
yarn install --pure-lockfile
```
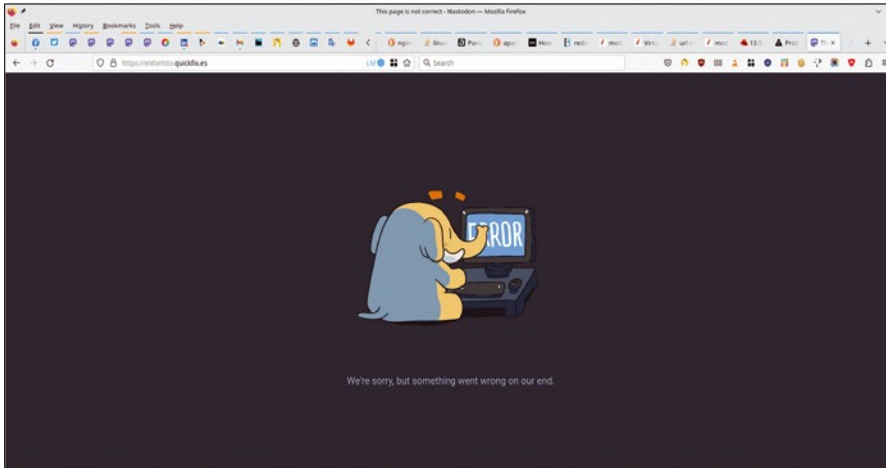
Figure 12: An angry mammoth banging a keyboard is really good news.

When you visit your page, you should see an error screen with an angry mammoth bashing a keyboard (Figure 12).

Don't panic! This is really good news. The error is there because you still have to start the Mastodon services. To get everything up and running, Mastodon starts three services: Sidekiq, for Ruby jobs Mastodon has to run in the background; Streaming for the feeds; and the Web service.

To start the services, use systemd. Copy over the Mastodon service files to systemd's service folder:

```
cp /home/mastodon/live/dist/mastodon-*.service ↩
/etc/systemd/system/
```

and start and enable Mastodon's services:

```
systemctl daemon-reload
systemctl enable --now mastodon-web ↩
mastodon-sidekiq mastodon-streaming
```

You are done! Reload your Mastodon, and happy tooting.

Post Install Setup

You can now log in using the email you set up and the password that was provided during the Mastodon configuration step.

Visiting *Preferences* as admin will show you a bunch of settings not available to regular users. With your newly running

Listing 5: Mastodon NGINX Site File

```
.
.
.
server {
 listen 80;
 listen [::]:80;
 server_name [your domain here];
 root /home/mastodon/live/public;
 location /.well-known/acme-challenge/ { allow all; }
 location / { return 301 https://$host$request_uri; }
}
.
.
.
```

instance, probably the most important section is *Administration*, and, under that, the *Server Settings* subsection. There you can change the name of the server, define who can register, and configure other settings.

Another important consideration is the set of rules you apply on your site. You can define rules against harassment, hate speech, toots about billionaire owners of competing platforms, and so on, by visiting *Administration | Server rules*.

To make sure rules are followed, visit the *Moderation* section, where you will be able to check reports from users, set up accounts with special roles (moderator, admin, owner), block certain domains from federating with your instance, and more.

There are many more options, of course, but most are self-explanatory. That said, if you need more information, your best source is the Mastodon docs themselves [13]. Federating with other instances it is more or less automatic: Find a user on another server you would like to follow, for example *@linuxmagazine@fosstodon.org*, and follow them. Bam! You're federated.

Conclusion

Despite the length of this article, there is more, much more, to Mastodon. How could there not be? At this moment of writing, Mastodon is just shy of 9 million accounts and has added 50 thousand new users in the past 24 hours. The landscape is constantly shifting, and new features are continually added to the platform.

This article will get you off to a good start, but be prepared to explore Mastodon and the Fediverse with an open mind, because tomorrow you will find exciting new things to play with. ∎∎∎

Info

[1] Mastodon: *https://joinmastodon.org/*

[2] Cory Doctorow, "Tiktok's enshittification": *https://pluralistic.net/2023/01/21/potemkin-ai/*

[3] ActivityPub: *https://activitypub.rocks/*

[4] *Linux Magazine* on Mastodon: *https://fosstodon.org/@linuxmagazine*

[5] A curated list of Mastodon servers: *https://joinmastodon.org/servers*

[6] William Shatner (yes, *that* William Shatner) uncovers a flaw in Mastodon: *https://twitter.com/WilliamShatner/status/1218176903107895296*

[7] WPCode for WordPress: *https://wordpress.org/plugins/insert-headers-and-footers/*

[8] Pixelfed: *https://pixelfed.org/*

[9] PeerTube: *https://joinpeertube.org/*

[10] Mastodon mobile apps: *https://joinmastodon.org/apps*

[11] Fedilab Mastodon client: *https://f-droid.org/en/packages/fr.gouv.etalab.mastodon/*

[12] Fail2Ban: *http://fail2ban.org/wiki/index.php/Main_Page*

[13] Mastodon docs: *https://docs.joinmastodon.org/*

Fediverse TV

PeerTube, the Fediverse's video platform, offers a decentralized, open source way to watch videos and live stream your own content. We'll show you how to get started and even set up your own instance. *By Paul Brown*

The Fediverse community is building a parallel Internet based on ActivityPub, a W3C-recommended standard for social media, and PeerTube [1] is the Fediverse's video service for individuals, communities, and other organizations.

As with Mastodon, it would be understandable to think of PeerTube as a Fediverse drop-in for the popular closed, proprietary alternatives such as YouTube. And, sure, you can use PeerTube that way, but you would be ignoring its merits and how it can deliver video streaming that takes control from powerful corporations and puts it into the hands of users.

Running a service that delivers video on the scale of YouTube requires an immense amount of bandwidth and storage, unless you decentralize the whole thing. To be clear, PeerTube does not offer an amount of media on the scale of YouTube. However, PeerTube does offer an ingenious way of growing the resources it needs along with the amount of media it serves. By using peer-to-peer (P2P) technology, PeerTube shares the load over *instances* (servers run by PeerTube community members). Counterintuitively with instances, the more viewers a video has, the lighter the load on the server where the instance is hosted, because the load is spread over more nodes in the network. In addition,

PeerTube's federated nature makes it possible for one instance to offer its visitors a much larger catalog of videos than it could if it were isolated and relied exclusively on its own storage.

In this article, I'll show you how to find a PeerTube instance to join, set up an account, live stream video, and even run your own instance.

Be a Viewer

Indeed, when you visit a PeerTube instance for the first time (Figure 1), unless the owner has changed the default configuration, you will see the most popular videos trending on the federation. It is entirely possible that none of these videos are being hosted on the instance you are visiting, but you can still watch them from that instance.

To see what is hosted on the instance itself, click on *Local videos* in the sidebar on the left. A click on *Discover*, on the other hand, will show a selection of videos from the federation split into categories. *Trending* is the default view previously mentioned, and *Recently added* shows you new videos from across the Fediverse.

If you want to know more about the instance you are visiting, click *About* at the bottom of the sidebar. The About page
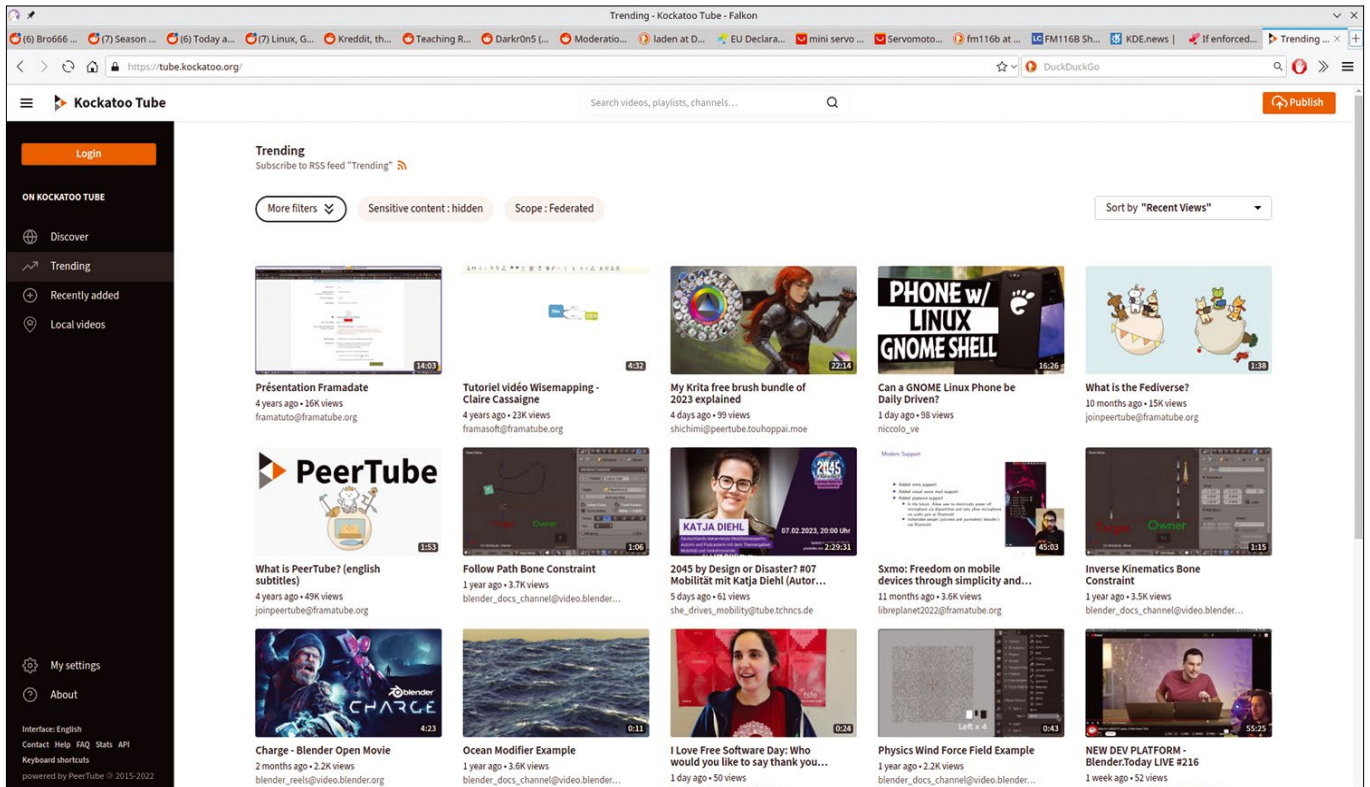
Figure 1: A PeerTube instance will display by default the most popular videos trending in the federation.

tells you why the maintainers created this instance, what the rules are, and, on the right, the instance's technical specs. Scrolling down, you will also be able to browse the instance's statistics (Figure 2), including how many users and videos are hosted on this site, among other details.

Knowing what sort of content the instance publishes, the rules and intentions laid out by the creators, and the size of the local library can give you an idea of whether you would like to join the PeerTube network via that instance or not.

Choosing a good instance to get started with is not hard, but it does require a bit of research. As with other Fediverse services, if you find the content unsavory, you may want to look elsewhere. Besides, a PeerTube instance that publishes extremely controversial content (wild conspiracy theories, not-safe-for-work content, racist rants, etc.) could find itself cut off from large chunks of the Peer-Tube network when other instances refuse to federate with them. If your account is on that server, you will be cut off too.

While you may think that a large

instance is a good choice, more users and content makes an instance harder to police, and, therefore, at more risk from spam, trolls, and unsavory content. A large library may also mean that an instance's maintainers use lax criteria for approving accounts, which, again, may lead to less-than-ideal content.

To help you choose, PeerTube offers an instance searcher [2]. You can search as a viewer or video maker and filter the instances by topic. The searcher also tells you how much space an instance gives you, the preferred language, if the instance offers live streaming, whether the submitted videos are moderated before publication or not, and more.
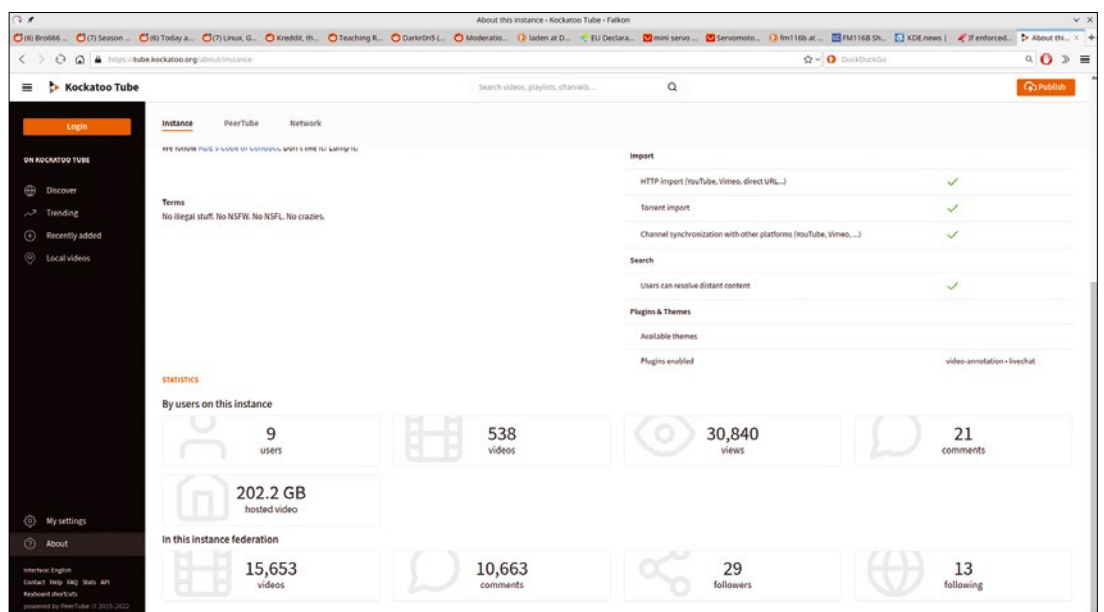
Figure 2: Kockatoo Tube is a small instance, with nine users and some 500 videos.
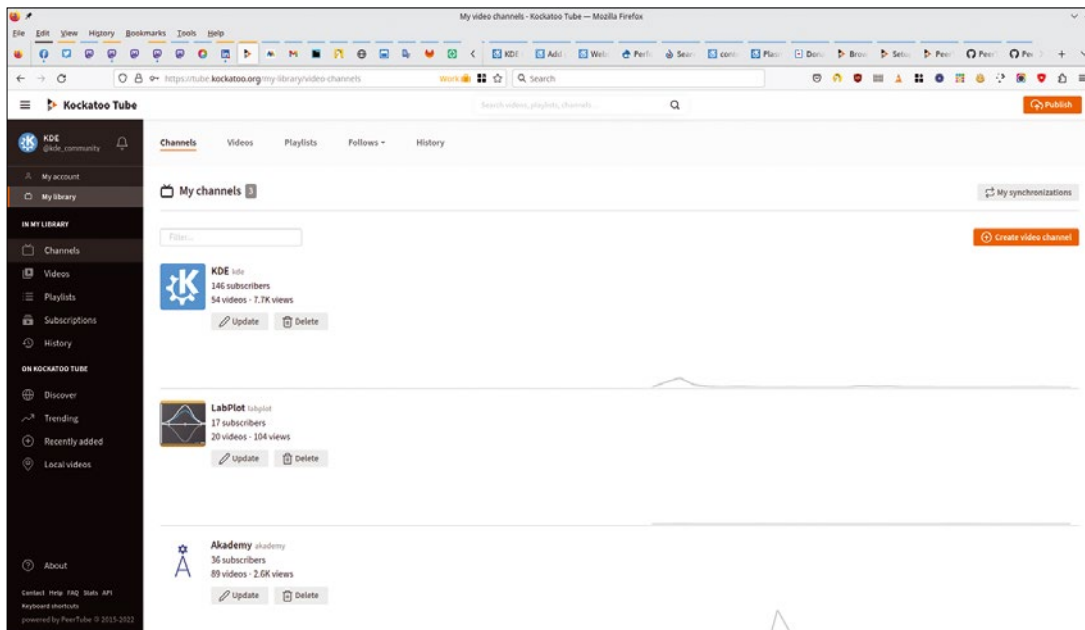
Figure 3: The *kde_community* account maintains three distinct channels: KDE, LabPlot, and Akademy.

Your Account

Once you have picked and registered with an instance (which follows the typical sign up or request to sign up/get confirmation email/log in pattern), you can set up your account. Again, this step is quite standard: Visit *Account | Settings* to set an avatar, display name, description, notification preferences, and more.

PeerTube differs slightly from other user-oriented video services: Each PeerTube account can manage several channels (Figure 3), each with a distinct look, description, libraries, and web address. As with its corporate-owned counterparts, Peer-Tube also lets you further classify videos into playlists.

Publishing a video is straightforward. While logged in, hit the *Publish* button in the upper right-hand corner of the Peer-Tube interface and select the video you want to upload or drag-and-drop a video onto the up-load area.

If you maintain a channel on YouTube, for example, many instances offer the possibility of im-porting videos from there. Hit *Publish*, click on the *Import URL* tab, and input the link to the video,

and PeerTube will import the clip, de-scription, and even the thumbnail. Fur-thermore, some in-stances offer the ser-vice of monitoring a YouTube channel and automatically importing any new videos that land on it. If you want to maintain channels both on YouTube and PeerTube, this makes it very conve-nient, as you only have to post to You-Tube and PeerTube will do the rest.

To take advantage of this feature, go to *Channels* in the left sidebar, and click on *My synchronizations* in the upper right-hand corner of the page. On the page that opens, click on *Add synchronization* and add the address of your YouTube channel. Then, indicate the PeerTube channel where you want the imported videos to land. If you are just starting out, you may want to choose the *Import all and watch for new publications* option, because that option will import all of your current videos and then start monitoring for new ones. Otherwise, you can choose *Only watch for new publi-cations*, which will only import new videos as they are pub-lished. When you are done, click *Create* and your instance
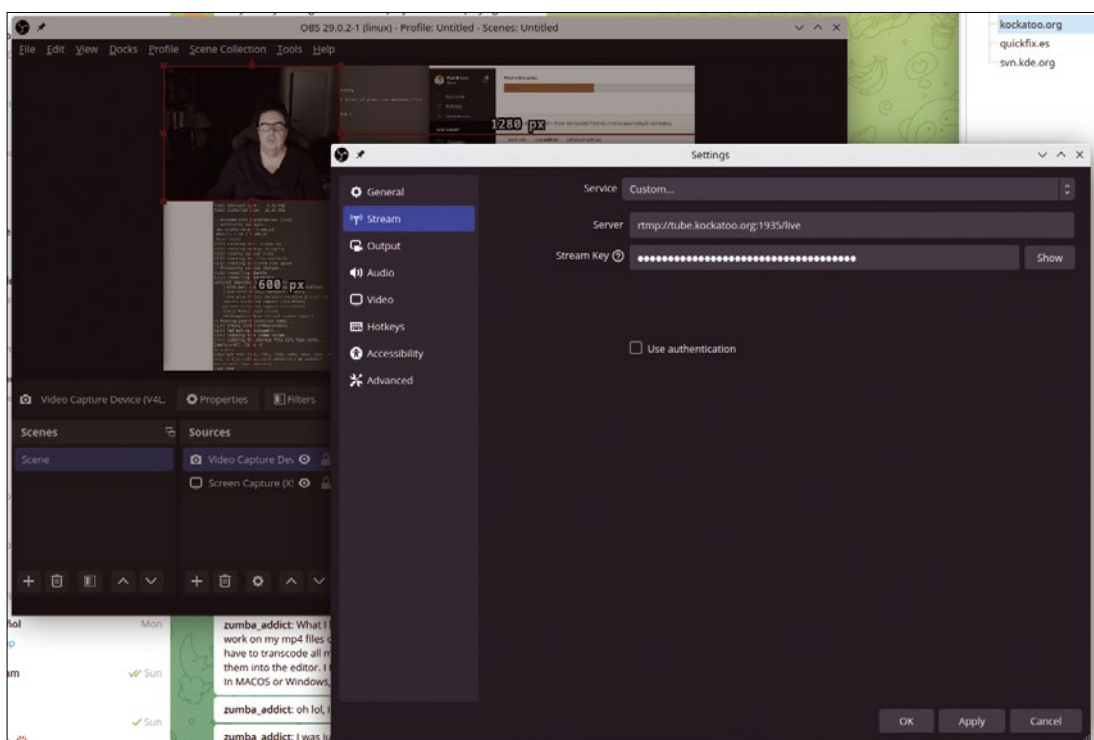
Figure 4: Setting up a live stream to PeerTube from OBS.

will start to follow the channel on the other service and will import videos from there.

The *Import with torrent* tab does what it says on the box: It lets you upload a video that is being shared over a torrent. Upload the torrent file from your local storage or paste a magnetic link and the video will appear after a while.

Streaming

You can also create content via live streaming. Choose the *Go live* tab, and select *This is a normal live* if your live stream will be a one-time event or *This is a permanent/recurring live* if you plan to make this your regular streaming platform.

If you choose a one-time live stream, the key PeerTube provides you for the stream is good for one live session. If you use the key again, it will overwrite the earlier recording. If you choose a recurring live stream, each time you start live streaming, even with the same key, PeerTube will generate a different recorded video.

Click on the *Go live* button and configure the stream as if it were a regular video by setting its title, a description, tags, the channel where it should be published, and so on. The *Live settings* tab is where you can find the live RTMP URL and the key that you will need to feed to your streaming software. In Open Broadcaster Software (OBS), for example, you will find the options under *Settings | Stream*. Select *Custom* from the *Service* drop-down list and then add the RTMP URL to the *Server* text box, and the key to the *Stream Key* password box (Figure 4).

Advanced settings allows you to add a thumbnail, which appears when the stream is not active or when the recording joins your library of prerecorded videos upon completion. *Plugin settings* lets you establish which plugins will be available to viewers during the stream. For example, many PeerTube instances provide a very decent chat plugin [3], which allows viewers to interact with the streamer (Figure 5).

Your Own Instance

Setting up your own PeerTube instance and then federating it is not terribly hard. First off, you do not need particularly high-end hardware [4], although a decent network connection is a must. With these requirements, most mid-range servers with decent-sized storage that you can rent from a hosting service for EUR30 (~ $32) will do.

The first installation step, as usual, involves securing your server. How to secure your server is beyond the scope of this

article. If you have any user-facing service, such as a blog, you already should have secured your server anyway. Notwithstanding, if you need some hints, look into a decent firewall, close every port except the ones you need open, configure your server to not honor password-enabled logins over SSH, and install and set up Fail2Ban [5].

For anyone who has set up a web service before, building a PeerTube instance follows the familiar steps of installing dependencies, setting up databases, installing and configuring PeerTube itself, and setting up your web server.

Written mostly in TypeScript, PeerTube relies on Node.js as its application server, Yarn as a package manager, and PostgreSQL and Redis to store data. FFmpeg is used for processing video in the background, and the supported web server is NGINX. The PeerTube documentation provides a guide on how to install these packages on your machine [6].

The official documentation also describes the process of setting up a *peertube* user on the system, setting up databases, downloading the PeerTube instance code, and tweaking the code to meet your particular needs. The documentation also covers how to set up your NGINX server, set PeerTube to run as a service, and, finally, how to log in as the administrator [7].

Administration

To quickly and easily acquire content for your instance, federate your instance with other instances. Fill in the details regarding your server so other admins can confirm that your instance is legit: Click on *Administration* in the sidebar, and go to *Configuration | Information*.

In the *Administration* section under *Configuration | Basic*, you can set up the policy for registering users, as well as how much space you are going to assign each user. Note that there is nothing wrong with maintaining a small, invitation-only instance. It is better to run an instance for 20 people, give them ample space for their content, and keep spammers at bay, than it is to have 2,000 accounts, very limited space for each user, and a mismanaged mess of an instance.
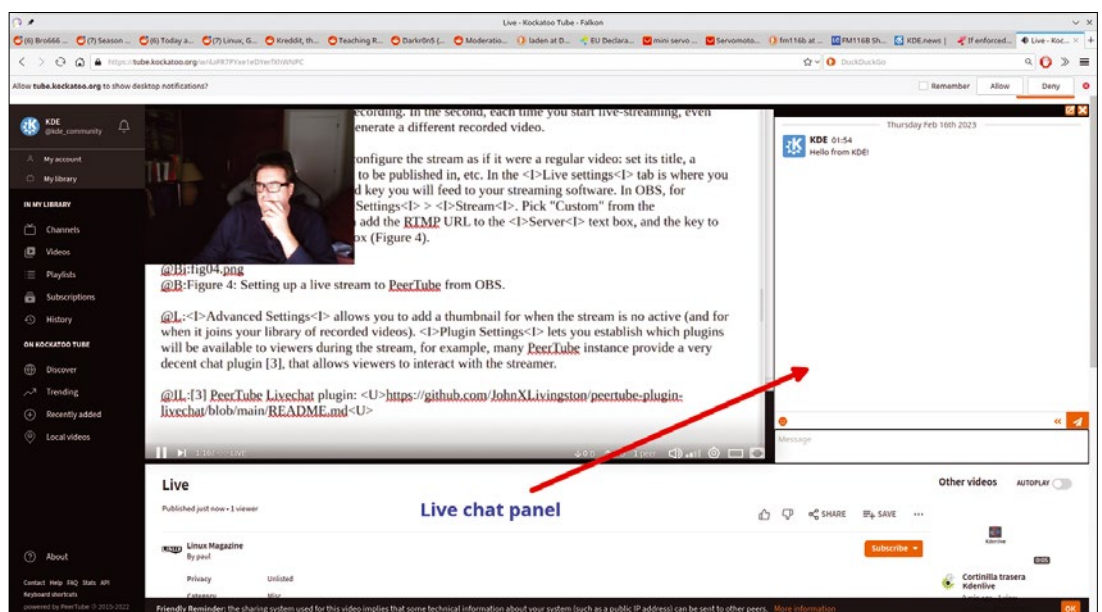

Figure 5: Streaming to PeerTube: Notice the live chat panel to the right of the stream.
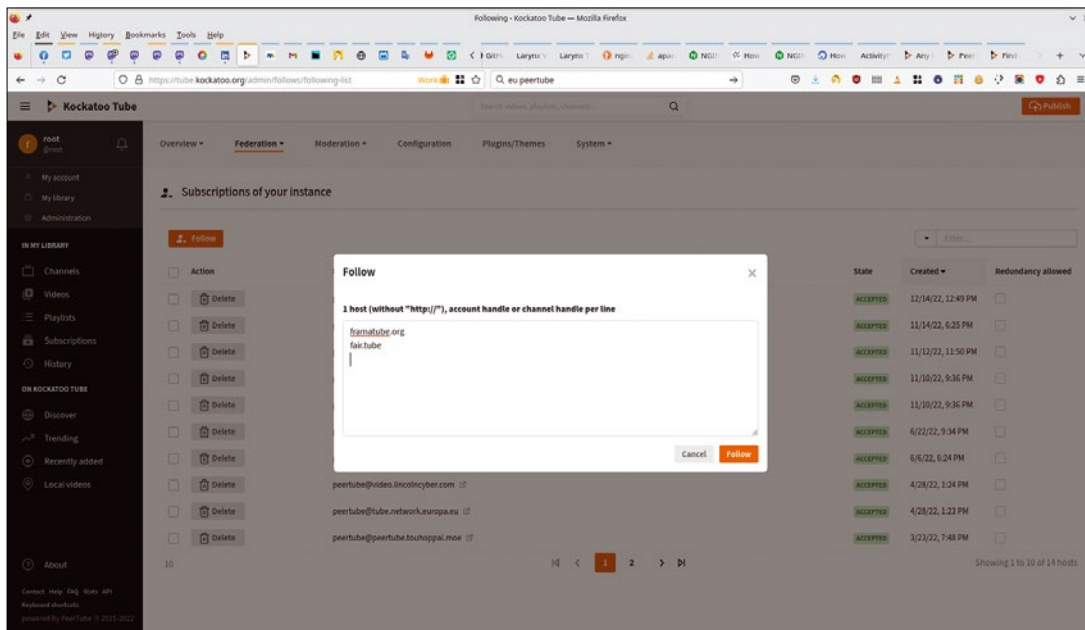
Figure 6: Add new instances to follow and show their content in your instance.

From *Basic*, you also can set how you want to federate your instance with others in the *Federation* section. Federating in PeerTube entails setting your instance to follow others – which means your instance will show and share the other instances' content (but not necessarily the other way around, unless they follow you back) – allowing other instances to follow you, and sharing your content on their instances.

To follow an instance with interesting content, go to *Federation | Following*, click on the *Follow* button and fill in the text box, with one instance address per line (Figure 6). For followers, you will receive notifications of instances that want to follow yours. You can approve or reject requests by visiting *Federation | Followers*.

Such is the nature of federation that videos may appear on your instance from instances that you are not directly following. These videos will be reaching you second hand, so to speak, because you are following an instance that is following them. Your instance will be helping to spread these videos from a third party in turn.

In general, this is a good thing, but if you find unsavory content appearing on your instance, you can block whole instances by going to *Moderation | Muted servers* and adding the offensive domain to the list. However, use this feature sparingly: PeerTube does not host any full videos from other instances on your instance, but it does count on federated instances sharing fragments of videos in order to spread the load in a true P2P fashion. PeerTube relies on instances federating with others to be able to provide a comprehensive service.

The *Moderation* tab also lets you control your instance's content on a more granular level, allowing you to mute accounts, remove rule-breaking comments, and check reports of abuse coming from users.

If you want to provide more services to your users – for example, the chat option on live streams mentioned earlier – check out the *Plugins/Themes* tab. Click on *Search*, and you can install plugins directly into your instance from there. Each

plugin has a complete page for settings. The live chat plugin, for example, lets you configure which rooms it will open by default, whether chats will be logged, and more.

Updating
From time to time, you will need to update your instance. Your instance will notify you when a new version of Peer-Tube is available. In general, you can follow the instructions available from Peer-Tube [8]; usually an upgrade is as simple as running a script.

That said, occasionally some vital component will change and running a script will not be enough. *Always* remember to check the new version's release notes before upgrading. The release notes will contain instructions that will help you along the way.

What's Next
Despite still being quite unknown, PeerTube is already a very mature, production-ready video platform with minimal bugs. It just needs its own "Elon Musk moment" to go mainstream.

That said, as this is the Fediverse, PeerTube integrates cleverly and seamlessly with other services. You can follow a Peer-Tube channel, for example, directly from Mastodon. Everything published to that channel will show up on Mastodon as a regular post with an attached video, and replies to the post on Mastodon will show up as comments on PeerTube. Perhaps with the rise in popularity of Mastodon, PeerTube will also become a popular service. ∎∎∎

Info
[1] PeerTube: *https://joinpeertube.org/*

[2] PeerTube instance searcher: *https://joinpeertube.org/instances*

[3] PeerTube livechat plugin: *https://github.com/JohnXLivingston/peertube-plugin-livechat/blob/main/README.md*

[4] Hardware requirements for an instance: *https://joinpeertube.org/faq#should-i-have-a-big-server-to-run-peertube*

[5] Fail2Ban: *https://www.fail2ban.org/wiki/index.php/Main_Page*

[6] Installing PeerTube dependencies: *https://docs.joinpeertube.org/dependencies*

[7] Installing PeerTube: *https://docs.joinpeertube.org/install-any-os*

[8] Upgrading PeerTube: *https://docs.joinpeertube.org/install-any-os?id=upgrade*

A decentralized photo sharing platform

Picture This!

Pixelfed offers an interesting alternative to centralized, algorithm-driven, commercial photo sharing services. *By Dmitri Popov*

Sometimes, it's painful to watch people make the same mistake again and again. When Flickr's bright star started to fade, anyone remotely interested in photography moved to Instagram. With Instagram losing its luster faster than a mountain hare loses its winter coat, there is a rush to VERO, Glass, and other photo sharing services that promise to be different but are essentially the same. The features offered by the current batch of Instagram challengers may vary, but the overall premise is unchanged: a service run by a commercial entity that dictates the rules and to whose fortunes and whims you're beholden. So you'll be forgiven for sagely shaking your head and murmuring to yourself, "Will they ever learn?"

Fortunately, shutterbugs and serious photographers who are not willing to go down the same road again can choose an alternative path: Pixelfed [1]. If you haven't heard the name before, you're not alone. While Pixelfed has been around for a while, it has been following the same trajectory as Mastodon. Twitter going down in flames has sent people scrambling for alternatives, with Mastodon providing a perfect harbor for Twitter refugees. While none of the mainstream photo sharing services have suffered a misfortune of a similar magnitude, the seed of doubt has been planted: Perhaps sharing your photos and building a following using a centralized commercial service is not all it's cracked up to be after all. This is where Pixelfed (Figure 1) comes into the picture (no pun intended).

A Federated Service

If you're thinking that Pixelfed is like Mastodon for photos, you're not far off the mark. Pixelfed has the same underpinnings: It's an open source, federated service based on the ActivityPub protocol. The "federated" part often causes confusion among those unfamiliar with the term. Ironically, most of us use an established and mature federated service every single day without perhaps even realizing it. Although it might be a stretch to call email a federated service, the underlying idea is basically the same: No matter which email service provider you choose, you can still exchange messages with anyone using any other provider thanks to common protocols like IMAP and SMTP. That's how a federated service operates, too. Pick a Pixelfed instance, create an account (which even looks like an email address, e.g., *@me@pixelfed.social*), and you can follow anyone on any other Pixelfed instance, and other users can follow you.

Federated services share another trait with email. An email address is unique, but the uniqueness of a specific username is limited to the service provider. The same is true for federated services. You can be Bobby Bushtail with the *thelongtail* username on the *pixel.social* instance, but nothing prevents someone from setting up the *@thelongtail@pixelfed.de* account and using the Bobby Bushtail name. If you want to reach the right Bobby, you have to know which Pixelfed instance your Bobby is on. This may be a problem if you're an influencer with a large following. However, it's worth
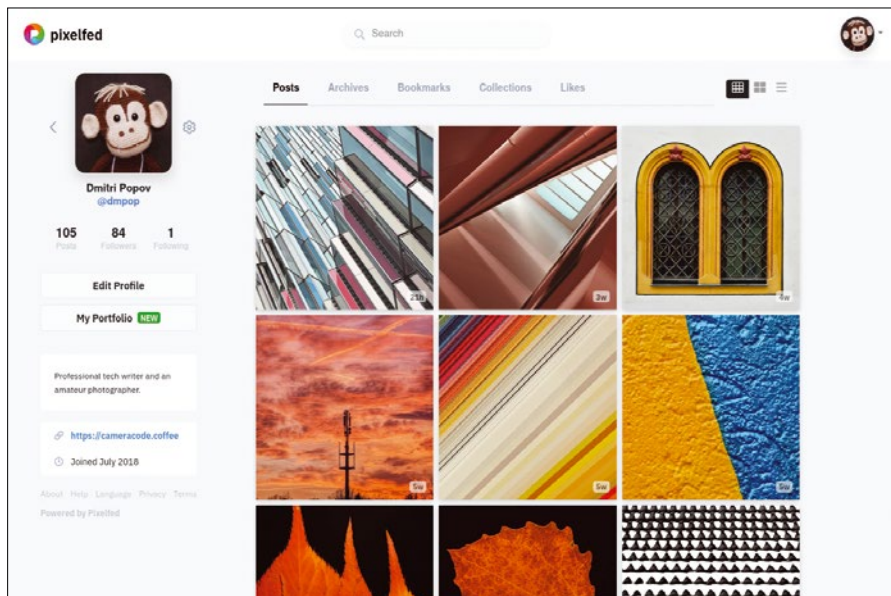
Figure 1: Pixelfed looks like a regular photo sharing platform.

keeping in mind that even though Pixelfed is often touted as an alternative to Instagram, it's not a drop-in replacement: Pixelfed is not suited for building a following and profiting from it. Whether this is an advantage or a drawback is up to you to decide.

Pros

The algorithmically managed timelines of mainstream social platforms are something nobody asked for and not many appreciate. If you prefer to see posts the way nature intended, you will definitely appreciate the fact that Pixelfed uses the chronological timeline for displaying your photos as well as posts from people you follow. No one can buy their way into your timeline, there are no ads, and no one is tracking you. You can interact with posts by liking them, commenting on them, and resharing them. You can also bookmark posts. But no amount of likes and reshares can push your post up a secret proprietary algorithm, because there is none. So while likes and reshares may give a tiny boost to your ego, that's pretty much all they do. Better still, if you don't care about likes and reshares at all, they can be disabled by Pixelfed instance administrators. Sadly, it's not something you can do as a regular user, but at least you can find a Pixelfed instance where this functionality is turned off by default.

Another aspect that sets Pixelfed apart from Instagram is its openness. Pixelfed doesn't lure you into a walled garden and force you to create an account just to be able to see photos. It also doesn't make life difficult for third-party apps and services by imposing arbitrary limits on its API. And, of course, Pixelfed makes it easy for you to move to a different instance or take your data out of service.

Cons

There are a few things you need to consider before you decide to make Pixelfed a home for your photos. Most Pixelfed instances are maintained by volunteers and not commercial entities with big bags of money (see the "Setting Up Your Own Instance" box). Consequently, volunteer efforts and donations

often are the only things that keep an instance running. This makes a Pixelfed instance's existence somewhat precarious. Worst case scenario, an instance disappears with no warning, without giving you a chance to migrate to a different server or export your data.

In addition, most Pixelfed instances have rather stiff storage limits. Forget about unlimited storage or free plans offering up to 1TB of storage. On most Pixelfed instances, you'll be lucky to get more than 10GB. Worse yet, you might not even be able to upgrade storage for an additional fee.

A Brief Intro to Pixelfed

Getting started with Pixelfed is as easy as it gets. Find the instance you like, create an account, and you're good to go. Sharing photos is not difficult either. Click *Create New Post*, and you can choose between three options. A regular post can be used to publish one or more photos, a story allows you to create posts that expire in 24 hours, and a collection lets you organize a selection of already published photos into a group.

When creating a regular post, you have the option to add up to 10 photos or videos. After choosing the desired files, Pixelfed immediately prompts you to add a caption, mark a post as sensitive, specify a license, disable commenting, etc. Pixelfed also makes it possible to crop and resize the added files as well as apply filters. Click on the left arrow in the upper-left corner of the dialog window to switch to the editing interface. Pixelfed offers several rather good filters that can spruce up dull-looking photos without much effort (Figure 2).

Setting Up Your Own Instance

One of Pixelfed's appealing traits is that you can set up your own instance. But even if you have the technical skills required to set up and maintain a Pixelfed server, the question is whether it's really worth it. The well-meant desire to offer others a place to share photos can quickly run into the stumbling blocks of reality. For starters, there are costs associated with running a platform that requires a lot of traffic and storage (as is usually the case with anything that involves sharing photos). The burden of ensuring uptime, backup, and maintenance also lies entirely on your shoulders. As always, the trickiest part is related to moderating human behavior. Even before you define a set of rules you want your users to abide by, you are at risk of going down a rabbit hole of trying to figure out answers to a myriad of questions. Will you allow sensitive content? If not, what are the criteria for labelling something as being sensitive? What conflict resolution mechanisms do you want to put in place? You may be tempted to figure out answers to these and many other questions as you go, but it's hardly a recipe for success. In short, running your own Pixelfed instance makes sense only if you have the means and spare time and you're willing to put effort into making your community thrive.

Grouping published photos into collections can come in especially handy for presenting photographic projects (Figure 3). Creating a collection is not difficult, but the process does have a quirk. When you choose *New Collection*, you're dropped into the *Create Collection* interface. Most of the options available here are self-explanatory, but the title of the *Add Posts* section is slightly misleading. You can add only one post, and that's it (this may be fixed by the time you read this). Once you've specified the available settings and published the collection, you can add as many photos to it as you like.

The *My Portfolio* feature is the most recent addition to Pixelfed. As the name suggests, you can use this feature to set up a portfolio containing a selection of photos you shared on Pixelfed (Figure 4). The portfolio features a different design, and it has a dedicated URL (*https://portfolio.pixelfed.social/ username*) that separates the portfolio from the rest of your Pixelfed account. This functionality makes it possible to use Pixelfed not only to share casual photos with your followers, but also showcase your best photographic work.

If you want to see photos posted by other Pixelfed users or you're looking for photographers to follow, Pixelfed has you covered. The *Local Feed* view shows recent posts from users on the Pixelfed instance you're on, while *Global Feed* pulls posts from other ActivityPub-based services such as Mastodon. Speaking of which, you're not limited to following only Pixelfed users: You can follow anyone on any ActivityPub-based service. This means that if you want to follow someone on *mastodon.social*, you can easily do it from your Pixelfed account. Simply use the search field to find the desired user by their username (e.g., *@username@mastodon.social*). The same applies to your Pixelfed account: Anyone can follow you from any other ActivityPub-powered service.

Finally, the *Discover* section offers several clever features. The *Daily Trending* section displays the most popular photos (Figure 5), which offers a way to find and follow photographers you like. The *My Memories* section displays photos taken on this day in previous years, giving you a chance to take a stroll down memory lane. You can view your most popular posts along with other useful
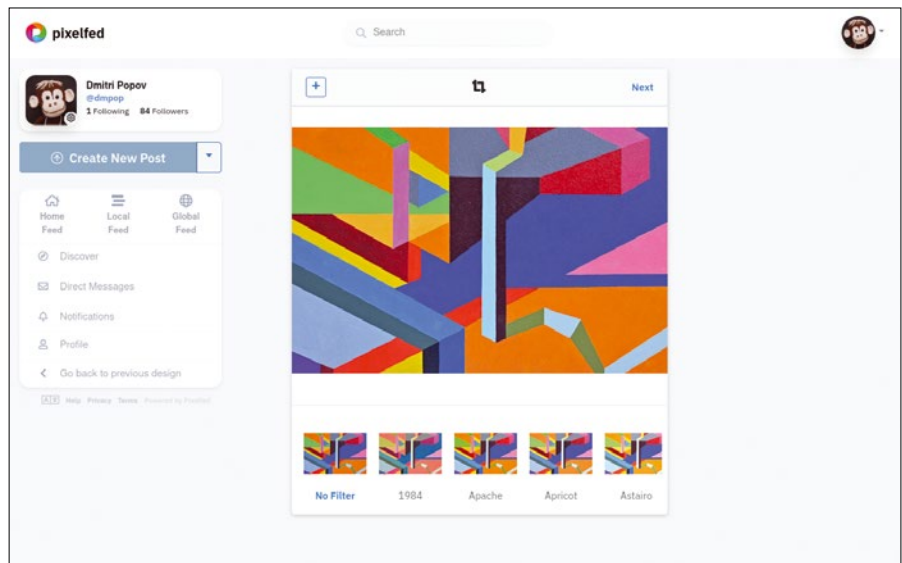
Figure 2: Pixelfed offers a decent selection of filters.
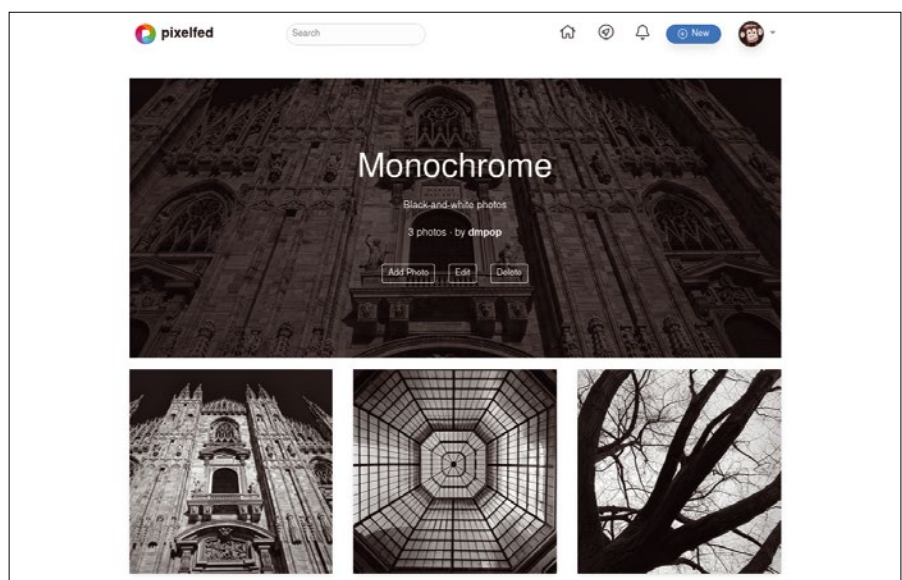
Figure 3: You can group photos into collections.
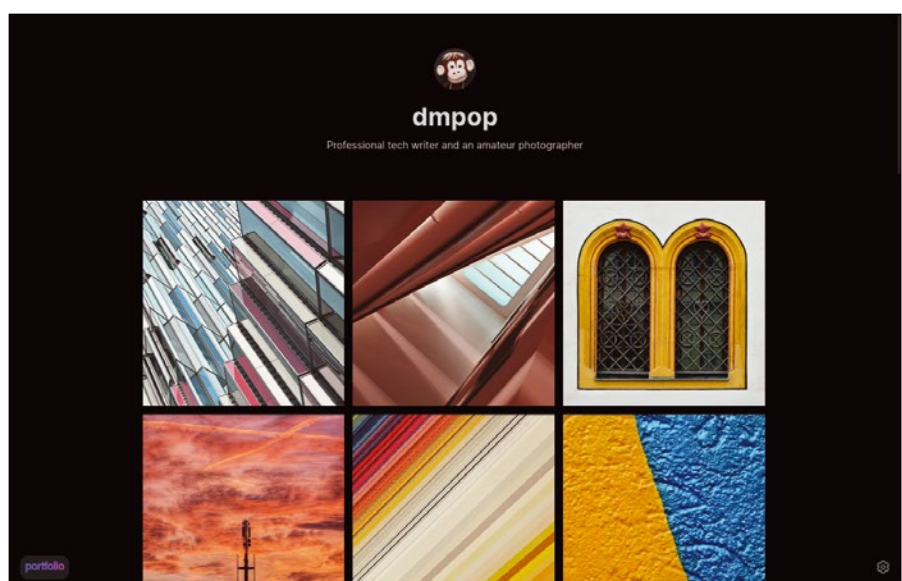
Figure 4: Pixelfed makes it possible to set up a portfolio.

stats in the *Account Insights* section, while *Find Friends* lists users that match your interests.

Closing Thoughts

Although Pixelfed has been around for quite a while, it still might feel a bit like a work in progress rather than a finished product. The current incarnation has done a lot of things right, and new features are popping up on a regular basis. But the fact that most instances offer only limited storage and are maintained by volunteers is something you need to consider. It still remains to be seen how successful individual Pixelfed instances will be at moderating as they grow in popularity. Opting for Pixelfed is not a zero-sum game, though. Nothing prevents you from using your current service while experimenting with Pixelfed. Even if you're not interested in using Pixelfed for sharing your photos, you might still want to use it to set up a portfolio to showcase your best work. In short, start slow and small, and see what works for you. ∎∎∎

Figure 5: Get a dose of inspiration and find photographers to follow with *Daily Trending*.

Info
[1] Pixelfed: *https://pixelfed.org/*

Author
Dmitri Popov has been writing exclusively about Linux and open source software for many years. His articles have appeared in Danish, British, US, and German magazines and websites. You can find more on his website at *cameracode.coffee*.

A customizable distro with better support

EuroLinux

EuroLinux offers enterprise class software with support from real engineers for a fair price. Bruce talks to the developers behind this RHEL-based distribution. *By Bruce Byfield*

Based on Red Hat Enterprise Linux (RHEL), EuroLinux [1] is a Polish-based commercial distribution that sells support from engineers and offers specialist editions, including a free community edition and editions for Raspberry Pi, containers, the cloud, and desktops. In keeping with the company's philosophy, my questions were answered by leading developers.

Linux Magazine (LM): Tell readers about the founding of EuroLinux.

EuroLinux (EL): EuroLinux was founded in 2013 by a group of individuals who had been selling and supporting

Author

Bruce Byfield is a computer journalist and a freelance writer and editor specializing in free and open source software. In addition to his writing projects, he also teaches live and e-learning courses. In his spare time, Bruce writes about Northwest Coast art (*http://brucebyfield.wordpress.com*). He is also co-founder of Prentice Pieces, a blog about writing and fantasy at *https://prenticepieces.com/*.

commercial open source software since 2000. Our company's name originates from the name of its first product, which was the EuroLinux operating system.

LM: Does the distribution have any connection to the campaigning organization [2] of the same name?

EL: We don't have a direct relationship, but we share many values. The emergence of our products is also in line with their goals, such as platform neutrality, restriction of patents in software, freedom of licensing, and no vendor lock-in.

LM: How does EuroLinux differ from other distributions based on RHEL?

EL: There are many elements that distinguish EuroLinux from other distributions both through our technical and business elements. We have our own EuroLinux Gaia build system, which is not dependent on *git.centos.org* (used mainly by the CentOS Stream and younger enterprise Linux distributions). As an organization, with the help of our build systems, we offer the

service of rebuilding and customizing the system for the customer, even in a closed environment. We are also the only clone that has a release (RPM release tag) of modular packages compatible 1:1 with RHEL. We are creating special versions of systems – publicly available ones include, for example, an image for Raspberry Pi or EuroLinux Desktop. EuroLinux Desktop is an extended version of the distribution designed for individuals and organizations that use Windows or macOS on a daily basis and are looking for a stable operating system, with a long and known life cycle, that looks like a Microsoft and Apple solution. [In addition], EuroLinux is offered as a complete solution. It is not divided into a series of paid elements. Therefore, the customer who purchases our product and support services does not pay for additional system modules such as HA, Load Balancer, or additional filesystems. We are also distinguished by a simple, friendly licensing system that does not limit the use of systems in virtual or cloud environments. The number of CPUs or cores does not affect the price of support.

LM: What are EuroLinux's goals and philosophy?

EL: The values that guide us are encapsulated in our motto:

- True product: We build the EuroLinux product stack based on proven, well-known, and widely used open source projects. As a result, we create enterprise-class software for use in the most demanding environments, ensuring stability and security. We provide and support complete, ready-to-use products that include standard modules to extend their core capabilities.
- Real Support: Our support division is made up of specialists with the highest certifications and the practical knowledge needed to provide professional customer service. The foundation of "real support" is facilitated through direct access to engineers and architects. Our aim is to keep support simple – a conversation between a real engineer and a client.
- Fair price: The price of our products does not depend on the number of processors in a machine. Nor do we impose licensing restrictions on the number of virtual machines running on a physical server. Nor do we make the customer dependent on our solutions, but allow them to freely choose their support and software provider. For us, the element of fair price is also a business model that maintains flexibility and openness to negotiation, including flat price unlimited subscriptions. That allows the customer to significantly reduce the costs associated with infrastructure upkeep.

User freedom is also a very important element of our ethos. Users can freely use the free public available version of the system and switch to the paid version at any time they choose. Conversely, they can use the paid, supported version and switch to the free version without any consequences. They will then receive only standard updates to the system, which are identical to the paid versions, including repository addresses.

LM: How specialized are EuroLinux's various products?

EL: We have nine products, all of which are directly compatible with well-known and popular solutions found in

enterprise environments and act as a drop-in replacement. In most products available to the public, one of the goals is to keep maximum compatibility with the upstream projects. This principle is quite similar to the Linux kernel's principle of not breaking userspace, even when, for example, some compilation flags or features are enabled or disabled. When it comes to the custom distributions, among the things that we change are bootchain, boot keys, additional kernel patches, different kernel versions, changes to the cryptographic policies and used algorithms, additional software, software removal, custom repositories, and more. It depends greatly on the customer's needs.

LM: What services does the company provide?

EL: The most notable services are our potential to build a customized Linux distribution tailored to our customers' requirements; the professional, unequaled support infrastructure; and our authorized training courses.

With the help of the proprietary EuroLinux Gaia tool, we enable customers to create dedicated Linux systems. Anyone can order their own Linux distribution from us, branded with a unique logo and sets of changes. We will produce it and maintain it (updates/custom patches) for up to 10 years.

Further, we support customers and users who are migrating their systems to EuroLinux. We provide scripts to automate the migration process and offer migration support. Additionally, technical support services are offered at three levels. Premium support is provided 24/7, and Standard support is provided 8/5. Basic support includes only installation support for the system or solution.

LM: How is EuroLinux organized? How are decisions made? How do the company and community interact?

EL: We have a board-led, team-based traditional organization. However, we operate a competency-based decision making ethos, where the technical decisions are made by those in the specialized technical team. Only company-wide or executive decisions are made at the board level.

EuroLinux's core team consists of a group of a dozen developers. Other employees of the company and group companies focus on work in project and product mode. A key element in building and maintaining (updates/patches/custom patches) is the heavy use of automation processes implemented in EuroLinux Gaia. This allows us to maintain agility and speed of system development and testing. At the moment we produce five basic versions of the system (EuroLinux 6 ELS, EuroLinux 7, EuroLinux 8, EuroLinux 9, and EuroLinux Desktop) and additionally several special use distributions.

We collect feedback from users and customers in a structured and regular manner, including through the partner channel. We also strive to seek out and realize responses to user requests. Each person has the opportunity to submit a proposal for change or their comments both through direct messages and openly on the GitHub platform. On the flip side, we engage through our social media and direct channels, our developments, and upcoming projects with our community.

At the beginning of each quarter, we publish our EuroLinux System Roadmap. In this post, we report on the work done over the last quarter, but most importantly it contains information about our development plans. Our roadmap largely reflects the needs reported by EuroLinux users.

LM: What features make the EuroLinux Desktop edition stand out?

EL: EuroLinux Desktop (Figure 1), like its server sibling, is based on the source code of RHEL 9. This means that it is compatible with RHEL, but it also includes additional functionality, extensions, and facilities.

The changes we've made primarily concern the system's appearance and usability. To this end, we used Gnome extensions. They come preconfigured in the system and are available immediately after installation, "out of the box." They significantly affect the perception of the software, but do not revolutionize it. All this is done to preserve the stability and security of the enterprise Linux family (RHEL, EuroLinux, CentOS), while making the system easier to use

for those accustomed to Windows or macOS. [For instance:]

- The translucent dock at the bottom of the screen is modeled after the latest versions of these systems. As the dock is permanently visible, in the EuroLinux Desktop we have disabled the now redundant overview mode and the Gnome hot corner function, which displays said mode when you mouse over the top left corner of the screen.

- EuroLinux Desktop allows icons to be placed on the desktop by default, although the popular Gnome window manager does not provide such functionality by default.

- When you right-click on the wallpaper, additional menu items appear (including *New Text Document*). We have also added icons for notifications from instant messaging or email programs. EuroLinux Desktop lets you switch day/night themes with a single click

on the moon icon in the dock and allows you to set a schedule for automatic theme switching based on the time of day.

We integrated the system with the Flathub repository. This provides easy access to more than 2,000 free apps. They are downloadable from the Software application. So you can install them with a single click – analogous to the Microsoft Store or App Store.

Conclusion

In many ways, EuroLinux is a case study in the position of commercial distributions. On the one hand, Euro-Linux carefully breaks down products and services for customers. On the other hand, EuroLinux remains continually aware of the expectations of the community. Not only does it continue to offer a download version, but its use of developers for support seems likely to appeal to open source advocates. It is a balancing act of which EuroLinux seems well aware. ∎∎∎

Figure 1: Eurolinux's Desktop edition. Photo courtesy of EuroLinux

Info

[1] EuroLinux (company): *https://shop.euro-linux.com*

[2] EuroLinux (activist organization): *https://en.wikipedia.org/wiki/EuroLinux*

**Get hiking suggestions
from your recorded tours**

Pathfinder

**When Mike Schilli is faced with the task of choosing a hiking
tour from his collection of city trails, he turns to a DIY
program trained to make useful suggestions.** *By Mike Schilli*

Which of the city walking routes recorded on the Komoot route planning service should I do again today? This is a question I ask myself surprisingly often. Depending on how I feel, I want the day's tour to be short or long, hilly or flat, and – accordingly – challenging or relaxing. Depending on time constraints, I might not want to venture too far from home.

My tours are recorded on the Komoot service, but it offers only very rudimentary filter options (Figure 1). In my case, using its smallest possible search radius of three miles, Komoot searches the entire San Francisco metropolitan area for saved tours rather than letting me narrow my search to individual neighborhoods. To enable more granular search criteria, what I have in mind for today's column is a command-line tool that uses the elevation profile, tour time, and distance to trail entry as filters to greatly reduce the selection.

The new tool can make its selection from tour data that is already available as GPX files on my hard disk from a previous

Snapshot column [1]. This XML format records the waypoints of the respective tour as geo-coordinates with elevation above sea level, including timestamps (Figure 2). In the absence of a publicly available API on Komoot's website, a web scraper in the previous column logged in to the site, fetched the GPX data, and copied it locally to the `tours/` directory on the hard disk using the numeric tour ID as a filename (as in `523799045.gpx`).

For post-processing, the CSV file in Listing 1 assigns easily recognizable route names to those IDs. What you see here is a selection of my tours, some of which are located in Germany and some

in the USA.

The rest of the conversion flow is now automatic. A preprocessor wades its way through the GPX data of all tours, determines their total duration, the elevation climbed, and the distance from home to the trailhead. Equipped with this metadata, a command-line program in Go later filters out the tours based on given criteria.

Dizzying Heights

Among other things, how strenuous a tour is depends on the meters of

Author

Mike Schilli works as a
software engineer in the
San Francisco Bay Area,
California. Each month
in his column, which has
been running since 1997,
he researches practical applications of
various programming languages. If you
email him at *mschilli@perlmeister.com*
he will gladly answer any questions.
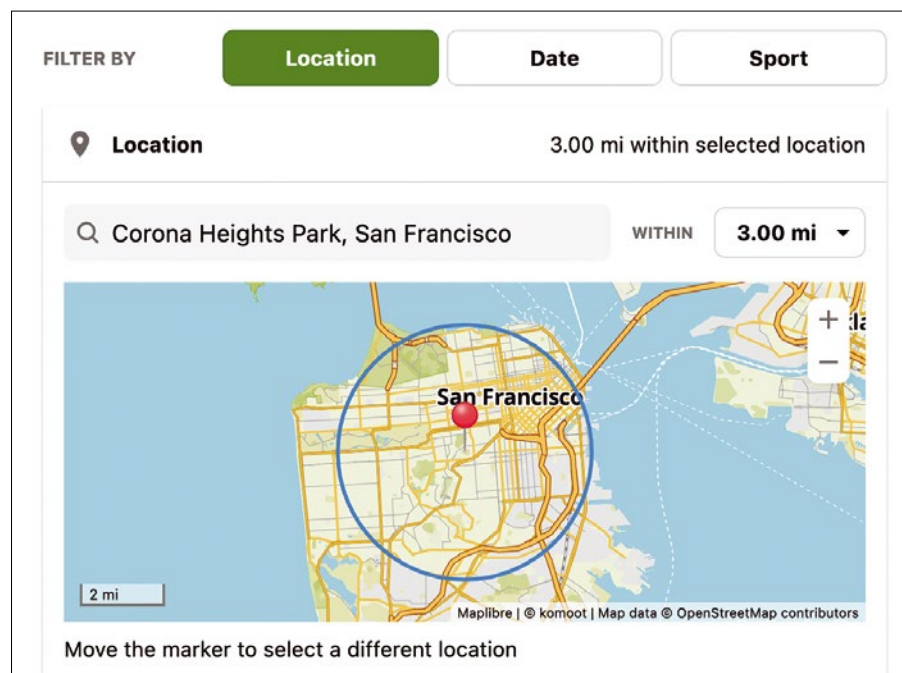
Figure 1: The Komoot route-planning service offers only rudimentary trail filters.

Lead Image © Sergey Mironov, 123RF.com

Figure 2: Example of a tour file in GPX format.

Figure 3: The dataframe is created from the GPX file's XML data.

altitude the hiker needs to walk uphill. Each waypoint in the GPX file (Figure 2) not only contains the geolocation's longitude and latitude (`trkpt lat/lon`), but also the current elevation above sea level in meters (`ele`). So, on a rising route, the height value will increase from point to point.

In order to compute the length of all uphill sections in meters above sea level for the tour, the algorithm needs to run through all the waypoints of the route, determine the difference in elevation to the following point by subtraction in each case, and finally add up these differences. Negative values are filtered out beforehand, because only the uphill gradients make the route more strenuous, not the downhill sections.

Listing 2 elegantly solves this task with just a few lines of R code. Installing R with your distribution's package manager, for example, on Ubuntu via

```
sudo apt-get install r-base-core
```

gets you the interpreter pretty quickly, but it does not install the GPX library needed by the program. To install the GPX library on your local machine

from the Comprehensive R Archive Network (CRAN), type

```
install.packages('gpx')
```

in an interactive R session (just call `R` at the command line). After doing so, the search path contains the `Rscript` program, which the listings in this column call from their shebang lines at the start. `Rscript` runs the individual listing's code through the R interpreter, so you can simply call the program name from the command line (like `./climb.r` in the case of Listing 2), after giving it executable permissions.

Let's look at the code: The `read_gpx()` function called in line 3 comes from the previously installed `gpx` library and expects the path to a GPX file. If the call is successful, the return value is a hodgepodge of named data containers and an array with tracks, which I cunningly named `tracks`. A GPX file can contain several of these, but only the first one is needed here. Line 4 retrieves the corresponding dataframe with the `hike$tracks[[1]]` expression (array element numbers in R start at one and not at zero) and assigns it to the `track` variable.

Figure 3 shows the data of the dataframe stored in the `track` variable. Because the elevation values in the `track` dataframe are in the `Elevation` column, line 5 extracts them with the expression `track$Elevation`. The script then assigns this vector with all the elevation values in the waypoints in the file to the `ele` variable.

Only Uphill Counts

Because the elevation values of the measuring points all lie in the `ele` vector, R's built-in `diff()` function determines the individual differences between them. The result is again a vector. If `ele` had the values (2,10,8,12), `diff()` would

Listing 1: tour-names.csv

```
id,name
401269499,Forest Hill Parkside Farmers Market
411337149,Presidio
394385030,Vulcan Stairs and around Buena Vista
526430358,Aggenstein
434310884,Heidelberg Schlierbach
418406673,Tank Hill Mt Olympus
514603221,Bernal around the Hill
434601991,Heidelberg Philo-Altstadt-Schloss
510083576,Forest Hill Stairs Mini Loop
405638419,Around Mt Davidson
393675355,Laidley Glen Park Loop
416081317,Sanchez-Mission
```

Listing 2: climb.r

```
01 #!/usr/bin/env Rscript
02 library("gpx")
03 hike <- read_gpx("tours/686129674.gpx")
04 track <- hike$tracks[[1]]
05 ele <- track$Elevation
06 steps <- diff(ele)
07 upsteps <- steps[steps > 0]
08 print(sum(upsteps))
```

Figure 4: The CSV file generated by `preproc.r` **contains the metadata for all the tours.**

make `(8,-2,4)` out of this. The recode statement in line 7 filters out the negative values, leaving only `(8,4)` in the example. The `sum()` function in line 8, which is also from the standard R library, grabs this vector and adds up its individual elements. In the example, the result would be 12.

The script in Listing 2 can be called from the command line. It outputs the sum total of meters of altitude climbed during the tour as an integer to the standard output. However, the filter program shown later does not just need the meters of altitude for one tour in the collection, but the values of all tours. In addition, besides the altitude meters, it also requires the latitude and longitude of the track's starting point and the duration of the tour in minutes. The preprocessor shown in Listing 3 provides the metadata for all the tours and creates a CSV file following the example in Figure 4.

How does the preprocessor work now? First, Listing 3 reads the CSV data from `tour-names.csv` (Listing 1) and fields a dataframe with the `id` and `name` columns. The `for` loop starting in line 4 iterates through all rows of this dataframe. Line 5 extracts the tour's numeric `id`, and line 6 compiles the path to the GPX file on disk from the id. The `read_gpx()` function then reads the tour data in GPX format from the downloaded file, and the remaining altitude calculation is analogous to Listing 2. Next, I need to append the computed numerical elevation change value to the dataframe in a new column named `ele`.

More Columns

To add a new column to a dataframe, you just need to assign a value to it. To do this, you can use either the dollar

notation (`idnames$newcol`) or the index numbers for row and column as in `idnames[row,col]`, where `col` is the new column number.

In the case at hand, `row` is the index number of the data series currently being processed by the `for` loop, and `col` is equal to `3`, because we want `"ele"` to be the third column in the dataframe. Assigning the sum value to `idnames[row,3]` will append the new column, but to make sure that it is given a name and not just new values, I also need to modify the `names(idnames)` array in line 14 by appending an element with the new column name.

It is important to insert a new column value before assigning a name to it, though, so that R knows that the dataframe has grown. After doing this, and only then, the name can be stored in `names`. Any attempt to do this beforehand throws an error message because R thinks that the dataframe is not big enough.

Starting Point

For the contents of the next two columns, with index numbers `4` and `5`, the R script looks for the latitude and longitude of the tour's starting point. Because the GPX data is available as a dataframe, this is a piece of cake. R just

addresses the first row with an index of 1 and uses the column name as the column index. This means that line 16 only has to ask for `[1, "Latitude"]` in the GPX dataframe to get the latitude of the first waypoint as a numerical value. The process for the geographical longitude is the same; lines 18 and 19 insert new columns, numbered 4 and 5, into the resulting `idnames` dataframe.

I am still missing the duration of the tour, which is determined by the code section starting in line 20 and then inserted into the resulting dataframe. The duration is calculated from the difference between the last and the first timestamps in the GPX file. Line 21 fetches the first line as index number 1 along with `"Time"`, which is the value in the column with the timestamps. The last entry from the GPX dataframe is determined in line 22 by R's standard `tail()` function with a parameter of 1 (meaning you only want the last element). The time column extraction method is the analog to determining the start time.

Listing 3: preproc.r

```
01 #!/usr/bin/env Rscript
02 library("gpx")
03 idnames <- read.csv("tour-names.csv")
04 for (row in 1:nrow(idnames)) {
05   id <- idnames[row, "id"]
06   gpxf <- paste("tours/", id, ".gpx", sep="")
07   hike <- read_gpx(gpxf)
08   track <- hike$tracks[[1]]
09   # elevation
10   ele <- track$Elevation
11   steps <- diff(ele)
12   upsteps <- steps[steps > 0]
13   idnames[row,3] = sum(upsteps)
14   names(idnames)[3] = "ele"
15   # starting point
16   idnames[row,4] = track[1, "Latitude"]
17   idnames[row,4] = track[1, "Latitude"]
18   names(idnames)[4] = "lat"
19   names(idnames)[5] = "lon"
20   # duration
21   start <- track[1, "Time"]
22   stop <- tail(track, 1)[1, "Time"]
23   mins <- round(as.numeric(difftime(stop, start),
             units="mins"), 0)
24   idnames[row,6] = mins
25   names(idnames)[6] = "mins"
26 }
27 write.csv(idnames)
```

Beginning and End

R's `difftime()` function computes the difference between two timestamps. For a result in minutes, line 23 calls R's standard `as.numeric()` function with the `units="mins"` parameter. The return value is a floating-point number with fractions of minutes, which the standard `round()` function rounds to the nearest integer with a precision of `0` (zero decimal places). That takes care of the tour duration, and lines 24 and 25 insert the value in column 6 into the resulting dataframe under the `"mins"` column header.

Finally, `write.csv()` writes the whole enchilada in CSV format to standard output, which the user redirects to the `tour-data.csv` file, as metadata to later enable automatic and fast filtering. The Go `hikefind` program, which I will be explaining in a minute, grabs the results from the file and applies its user-configured filters. To do this, Listing 4 uses `readCSV()` to read the metadata into memory by placing the individual entries into an array slice with elements of the `Tour` type. Defined starting in line 10, the elements of this type store all the important metadata, such as the duration, meters of altitude, and starting point.

As you can see and have probably expected, data processing in Go is far less elegant than in R. The *encoding/csv* package understands the CSV format, but Go's reader type needs to laboriously work its way through the lines of the file, checking for the end of file (line 29) and handling any read errors. Because the first line in the CSV format lists the column names, the logic starting in line 36 works its way past this with the `firstLine` Boolean variable.

Lines 41 to 48 then extract the numeric column values using `ParseFloat()` and `ParseInt()` along with the respective precision (32- or 64-bit) and a base of 10 for integers, followed by line 49 to set the corresponding attributes in the `Tour` type structure. Line 55 appends a single instance of this structure to the array slice with all the line data from the CSV file, and the action continues with the next round.

Choosy

The main program in Listing 5 understands a number of filter flags: `--gain` is a qualifying tour's maximum elevation gain in meters and `--radius` is the maximum distance from my home base, the coordinates of which are defined by `home` in line 9. Adjust this to your private settings for the best results. The command-line parameter `--mins` defines the

Listing 4: csvread.go

```
01 package main
02 import (
03   "encoding/csv"
04   "fmt"
05   "io"
06   "os"
07   "strconv"
08 )
09 const csvFile = "tour-data.csv"
10 type Tour struct {
11   name string
12   file string
13   gain int
14   lat  float64
15   lng  float64
16   mins int
17 }
18 func readCSV() ([]Tour, error) {
19   _, err := os.Stat(csvFile)
20   f, err := os.Open(csvFile)
21   if err != nil {
22     panic(err)
23   }
24   tours := []Tour{}
25   r := csv.NewReader(f)
26   firstLine := true
27   for {
28     record, err := r.Read()
29     if err == io.EOF {
30       break
31     }
32     if err != nil {
33       fmt.Printf("Error\n")
34       return tours, err
35     }
36     if firstLine {
37       // skip header
38       firstLine = false
39       continue
40     }
41     gain, err := strconv.ParseFloat(record[3], 32)
42     panicOnErr(err)
43     lat, err := strconv.ParseFloat(record[4], 64)
44     panicOnErr(err)
45     lng, err := strconv.ParseFloat(record[5], 64)
46     panicOnErr(err)
47     mins, err := strconv.ParseInt(record[6], 10, 64)
48     panicOnErr(err)
49     tour := Tour{
50       name: record[2],
51       gain: int(gain),
52       lat:  lat,
53       lng:  lng,
54       mins: int(mins)}
55     tours = append(tours, tour)
56   }
57   return tours, nil
58 }
59 func panicOnErr(err error) {
60   if err != nil {
61     panic(err)
62   }
63 }
```

Listing 5: hikefind.go

```
01 package main
02 import (
03    "flag"
04    "fmt"
05    "github.com/fatih/color"
06    geo "github.com/kellydunn/golang-geo"
07 )
08 func main() {
09    home := geo.NewPoint(37.751051, -122.427288)
10    gain := flag.Int("gain", 0, "elevation gain")
11    radius := flag.Float64("radius", 0, "radius from home")
12    mins := flag.Int("mins", 0, "hiking time in minutes")
13    flag.Parse()
14    flag.Usage = func() {
15       fmt.Print(`hikefind [--gain=max-gain]
                 [--radius=max-dist] [--mins=max-mins]`)
16    }
17    tours, err := readCSV()
18    if err != nil {
19       panic(err)
20    }
21    for _, tour := range tours {
22       if *gain != 0 && tour.gain > *gain {
23          continue
24       }
25       start := geo.NewPoint(tour.lat, tour.lng)
26       dist := home.GreatCircleDistance(start)
27       if *radius != 0 && dist > *radius {
28          continue
29       }
30       if *mins != 0 && tour.mins > *mins {
31          continue
32       }
33       fmt.Printf("%s: [%s:%s:%s]\n",
34          tour.name,
35          color.RedString(fmt.Sprintf("%dm", tour.gain)),
36          color.GreenString(fmt.Sprintf("%.1fkm", dist)),
37          color.BlueString(fmt.Sprintf("%dmins", tour.mins)))
38    }
39 }
```

maximum tour duration in minutes. The flags take either floating-point or integer values from the user, which `hikefind` converts to its internal types. `hikefind` then uses the values to whittle down qualifying tours from the CSV metafile.

Listing 6: Compiling

```
$ go mod init hikefind
$ go mod tidy
$ go build hikefind.go csvread.go
```

The `for` loop starting at line 21 iterates over all metadata read using `readCSV()` in line 17 and applies the three implemented filters: `gain`, `radius`, and `mins`. The distance from home is checked by the `radius` filter using the GitHub *kellydunn/golang-geo* package. This package uses the `GreatCircleDistance()` function to determine the distance between the two geo-points in kilometers and then compares the numerical result with the defined filter value.

If one of the three filters is tripped, the `for` loop `continues` with the next round without producing any output. But if an entry passes through all the filters unscathed, the print statement from line 33 outputs the tour.

Colorful and in Color

To make the meta values of the printed tours more eye-catching later on, Listing 5 initially includes the *fatih/color* package from GitHub, which provides functions to output the ANSI color codes commonly used by terminals.

You can compile Listings 4 and 5 with the usual three-card trick (Listing 6). The resulting `hikefind` binary either outputs all tours (Figure 5, without command-line options) or gives you a more compact selection using arbitrary combinations of the different filter types.

Figure 6 shows all the hiking trails within 10 kilometers of my adopted hometown of San Francisco that climb less than 100 meters in elevation and can be completed in one hour or less. Only three routes remain – it's a pretty hilly city after all. ∎∎∎

Figure 5: When called without options, `hikefind` lists all available trails.

Figure 6: Use the filter options to create a selection entirely to your liking.

Info

[1] "Go Retrieves GPS Data from the Komoot App" by Mike Schilli, *Linux Magazine*, issue 252, November 2021, *https://www.linux-magazine.com/Issues/2021/252/Plan-Your-Hike/(language)/eng-US*
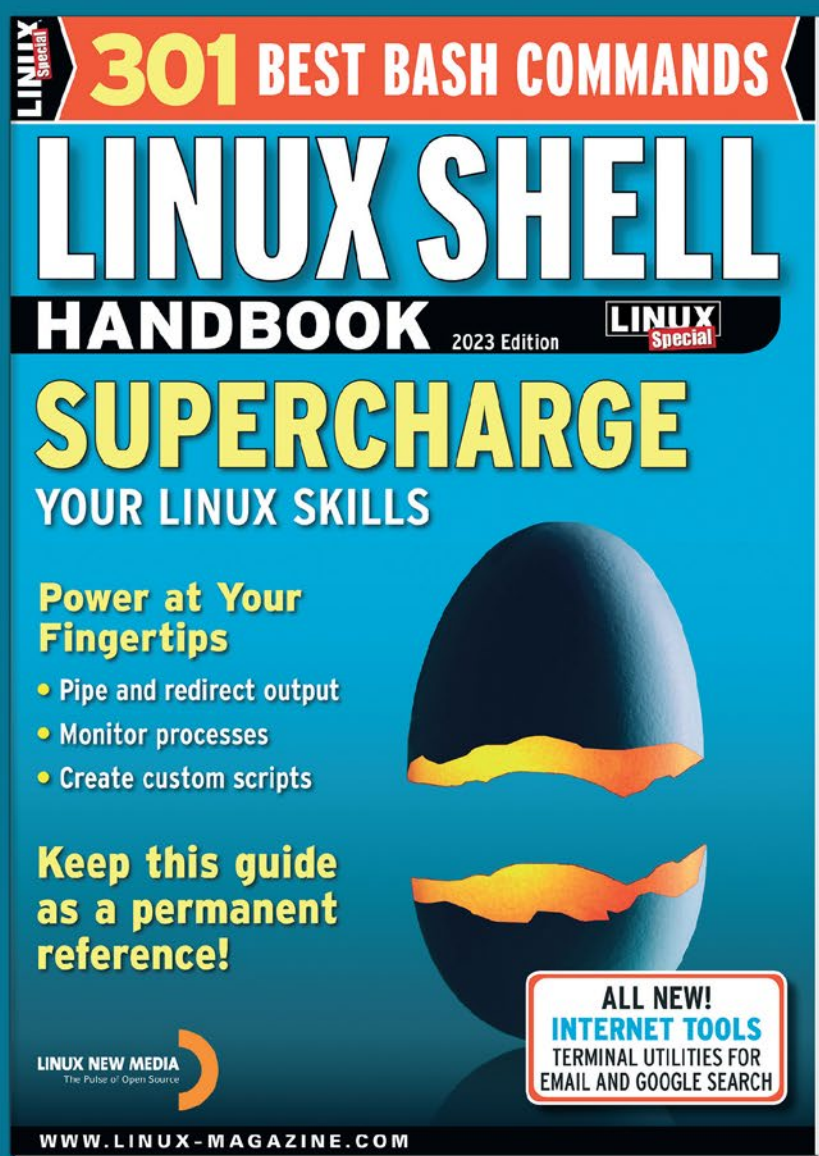
Using Mastodon
from the command line

toot

If you are looking for a Twitter alternative, toot lets you interact with Mastodon from the command line. *By Bruce Byfield*

An open source, decentralized version of Twitter, Mastodon has been in development since 2016. With Elon Musk's erratic decisions since purchasing Twitter in late 2022, many users predicting Twitter's doom have been searching desperately for alternatives. The result has been a new surge in interest in Mastodon – even if many newcomers remain puzzled by its open source organization. The interest is so great that Mastodon's founder Eugen Rochko reports that he has received offers of "hundreds of thousands of dollars" from five investors – offers he refused so as not to endanger Mastodon's non-profit status [1]. As I write, the exodus from Twitter appears to have slowed, but along with the renewed interest in Mastodon has come a renewed interest among Linux users in toot [2], an open source command-line client for Mastodon.

Toot takes its name from Mastodon's equivalent of Twitter's tweets, the microblog messages posted by users. Officially, the term is obsolete, with Mastodon's latest release replacing

toot with a simple *Publish* button [3], but the term remains widely used. Toot compares favorably with Twitter clients for Linux such as Twidge, Oyster, and Rainbow Stream, many of which are not up-to-date, and have an alarming tendency to come and go without warning. By contrast, although current users must still use the web interface to create a Mastodon account for everyday use, toot is a complete replacement for the web version of Mastodon (Figure 1), providing functionality that is both thoroughly

documented and up-to-date with Mastodon through a series of commands and options that keeps users' hands firmly on the keyboard.

Getting Started

You will find toot in the package repositories of many distributions and you can also install toot using Homebrew, although some sources are more current than others. However you choose to install, to use toot, you must create an account in a Mastodon instance from a web browser before logging in with the command `toot login_cli` (Figure 2). Upon launching, toot creates a configuration file in your home directory at `.config/toot/config.json`. You
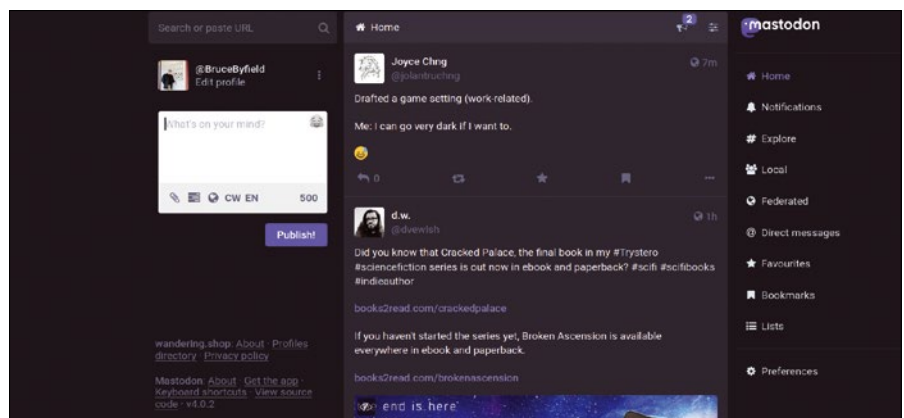
Figure 1: The Mastodon web application.

```
bb@ilvarness:~$ toot login_cli
Choose an instance [mastodon.social]: wandering.shop
Looking up instance info...
Found instance The Wandering Shop running Mastodon version 4.0.2
Registering application...
Application tokens saved.
Log in to wandering.shop
Email: bbyfield@axion.net
Password:
Authenticating...
Access token saved to config at: /home/bb/.config/toot/config.json

✓ Successfully logged in.
```

Figure 2: Logging in to toot.

```
bb@ilvarness:~$ toot post 'testing toot Mastodon client'
Toot posted: https://wandering.shop/@BruceByfield/109814375099334798
```

Figure 3: One of several ways to post a toot.

must then specify a Mastodon instance to log in to – but note that its name must be entered in lowercase characters with periods rather than spaces between the words, regardless of how the instance name is presented in the web interface. After toot locates the instance, enter your email and password to log in. For later access, toot stores your email and a temporary access token in the configuration file so that your password is never stored unencrypted. Once you are connected to an instance, you can enter `user whoami` to display user details and `toot whois` to display account details. To switch accounts, use `toot activate`. To see a list of active accounts, enter `toot auth`.

Posting Toots

You can post a toot in several ways. The simplest (as shown in Figure 3) is:

```
toot post 'MESSAGE'
```

Alternatively, you can compose a toot in a text editor of your choice (Figure 4) – Vim in this example – with the following command:

```
toot post --editor vim
```

Save and exit the toot as you normally would in the editor, and the toot is also posted. If you prefer to always use a text editor, define your preferred editor as an environmental variable (for instance, `export EDITOR=vim`), and then you can simply use the following command:

```
toot post --editor
```

With all these variations of the command, you can add up to four attachments. For the first attachment, use the option

```
-media path/to/image1.png ↗
  --description "TEXT" \
```

and then change the image number for the subsequent attachments. If necessary, you can flag attachments with:

```
--media nsfw.png --sensitive
```

Once a toot is posted, you can see any replies using `toot notifications`.

Table 1: Viewing Options

`toot timeline --public --local`	Public timeline, only this Mastodon instance
`toot timeline --tag photo`	Posts tagged with #photo
`toot timeline --count NUMBER`	Fetch up to 20 toots
`toot timeline --once`	Don't prompt to fetch more toots

Viewing the Timeline

When viewing toots in your timeline, the basic command `toot timeline` displays the most recent toot. It then gives you the option of either displaying the last dozen toots or else exiting the client altogether. You can also use the options listed in Table 1 to modify which toots are displayed.

At the bottom of each toot, you will find its ID and whether it has been reblogged. You can pin one of your own toots to the top of your timeline with `toot pin ID` or delete it with `toot delete ID` (Figure 5).

You can locate other people on Mastodon with

```
toot search "NAME"
```

or by their Mastodon instance address with

```
toot search NAME@INSTANCE
```

With an instance address, you can also use the self-explanatory commands `follow`, `mute`, `block`, or `unfollow`. To like a toot, used the command

```
toot favourite ID
```

(note the UK English spelling). To reblog toot use `toot reblog ID`.

Figure 4: You can compose a toot in the text editor of your choice.

Should you decide to undo any of these actions, you can run the same commands prefaced by un. For example, if you decide on second thought not to favor a toot, use the command:

Figure 5: The toot timeline.

Table 2: Curses Interface Navigation

k or up arrow	Move up the list of toots
j or down arrow	Move down the list of toots
h	Show a help screen
t	View status thread
v	View the current toot in a browser
b	Boost or unboost a status
f	Favorite or unfavorite a status
q	Quit the curses interface and return to the command line
s	Show sensitive content
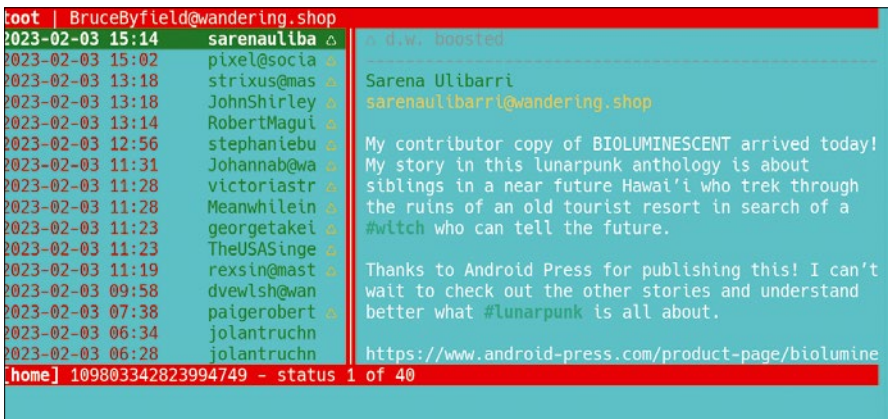
Figure 6: Toot includes an alternative curses-based interface for working with the timeline.

```
toot unfavourite ID
```

The rest you can probably figure out for yourself. If necessary, though, for detailed summaries, run

```
toot COMMAND --help
```

You can perform all these basic functions from the command line or start the curses user interface with toot tui (Figure 6). The curses interface is navigated by the keyboard commands shown in Table 2.

The Right Application

Although still in development, toot already stands out as a well-designed application. Its commands are logical, consistent, and well-documented, especially for a project still in the early stages of development. If you are unfamiliar with Twitter or Mastodon, you should have few problems learning the basics. If you are familiar with either, you may be able to guess the command you need with surprising accuracy. In fact, ironically, after working with toot from the command line, I found the curses interface more difficult to use. Considerable thought has gone into toot. In light of recent events at Twitter, it may just be the right application at the right time. ∎∎∎

Info

[1] Mastodon investment offers: *https://arstechnica.com/tech-policy/ 2022/12/twitter-rival-mastodon-rejects- funding-to-preserve-nonprofit-status/*

[2] toot: *https://toot.readthedocs.io/en/ latest/index.html*

[3] Mastodon retires "toot": *https://gizmodo.com/mastodon-toot- retired-twitter-tweet-equivalent- 1849786221*

FOSSLIFE

Open for All

**News • Careers • Life in Tech
Skills • Resources**

FOSSlife.org

An open source appointment manager

Scheduling Specialist

If you have a business that requires customers to make an appointment in advance for services, letting them request the appointment via Easy!Appointments can free up your phone line. *By Rubén Llorente*

H air salons, medical offices, and other small businesses require customers to book an appointment. Traditionally, small businesses manage these appointments by offering a phone number. Customers call in, and an employee (or the business owner!) sets a date and time, usually writing everything down in an appointment book.

While this approach works, it suffers from a number of shortcomings. Customers can only phone in when the office is open. In addition, internal management becomes cumbersome if more than one person needs to know the next day's schedule: The receptionist then has to spend time telling each professional the weekly schedule instead of working on more important tasks.

Table 1: Easy!Appointments 1.4.3 Prerequisites

Web server	Apache 2.4
Database	MySQL 5.7
PHP engine	PHP 7.3

LAMP Servers in a Nutshell

Linux, Apache, MySQL, PHP (LAMP) describes machines serving web applications built upon these tools.

Setting up a LAMP server is very easy, but deploying one is outside the scope of this article. For testing purposes, you can set up a LAMP server in the cloud using a virtual private server (VPS) from Hetzner with a LAMP stack preconfigured (Figure 1). For more information, see Hetzner's documentation [2].
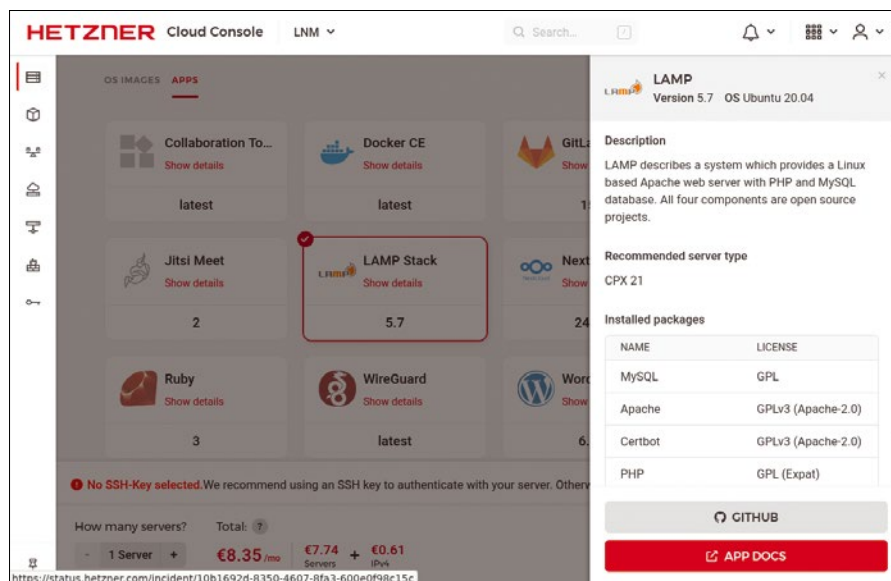
Figure 1: You can deploy a LAMP server via the Hetzner Cloud Console for testing purposes, if you don't want to roll one manually. © Hetzner Online GmbH

Photo by Leone Venter on Unsplash

To minimize (potentially significantly) these issues, you can leverage the power of technology and let computers do the hard work. Easy!Appointments [1], a free, open source web application, lets customers make appointments over the Internet even after business hours, while letting employees and business owners check their schedules from their smartphones without bothering the receptionist.

Installing Easy!Appointments

Easy!Appointments has a short list of prerequisites. It will run on any modern LAMP stack (see the "LAMP Servers in a Nutshell" box). Table 1 shows the current requirements for the latest stable version at the time of writing this article. I have also tested Easy!Appointments with OpenBSD's httpd and the MariaDB database with great success.

The instructions provided here are for setting up Easy!Appointments on a Debian LAMP server. If you are using a minimal Hetzner LAMP install, you will need to install some additional software:

```
apt-get update
apt-get install unzip php7.4-mbstring
systemctl restart apache2
```

Once you have a LAMP server configured, installing Easy!Appointments is a breeze. First, move to your web server root (e.g., /var/www/demo.operationalsecurity.es) and download Easy!Appointments:

```
cd /var/www/demo.operationalsecurity.es
wget https://github.com/alextselegidis/
  easyappointments/releases/download/
  1.4.3/easyappointments-1.4.3.zip
```

Next, decompress the downloaded archive (you will need the unzip program installed on your system):

```
unzip easyappointments-1.4.3.zip
```

Then, grant the web server user ownership over the files:

```
chown -R www-data ./*
```

You will now create a database for the web application as shown in Listing 1. If you are using a minimal Hetzner LAMP install, you will find MySQL's root password (needed to issue all of the following commands) at /root/.hcloud_ password. Keep in mind that passing SQL instructions to

Figure 2: Easy!Appointment's installation Wizard will ask you for some information in order to get the site set up.

Listing 1: Creating a Database

```
echo "CREATE DATABASE easyappointments;" | mysql -u root -p
echo "CREATE USER easyappointments IDENTIFIED BY 'password';" | mysql -u root -p
echo "GRANT ALL PRIVILEGES ON easyappointments.* to easyappointments;" | mysql -u root -p
```

Listing 2: Example config.php

```
01 <?php
02 class Config {
03
04   // ----------------------------------------------------
05   // GENERAL SETTINGS
06   // ----------------------------------------------------
07
08   const BASE_URL     = 'https://demo.operationalsecurity.es';
09   const LANGUAGE     = 'english';
10   const DEBUG_MODE   = FALSE;
11
12   // ----------------------------------------------------
13   // DATABASE SETTINGS
14   // ----------------------------------------------------
15
16   const DB_HOST      = '127.0.0.1';
17   const DB_NAME      = 'easyappointments';
18   const DB_USERNAME  = 'easyappointments';
19   const DB_PASSWORD  = 'some_password_here';
20
21   // ----------------------------------------------------
22   // GOOGLE CALENDAR SYNC
23   // ----------------------------------------------------
24
25   const GOOGLE_SYNC_FEATURE   = FALSE; // Enter TRUE or FALSE
26   const GOOGLE_PRODUCT_NAME   = '';
27   const GOOGLE_CLIENT_ID      = '';
28   const GOOGLE_CLIENT_SECRET  = '';
29   const GOOGLE_API_KEY        = '';
30 }
```

Table 2: Fixed and Flexible Appointments

Fixed	Appointments stack according to their defined duration. (If you have a 30-minute appointment at 09:30, the next available appointment will be at 10:00.)
Flexible	Appointments stack to the next interval of 15 minutes. (If you have a 30-minute appointment at 09:30, the next available appointment will be at 10:00, and the next one at 10:15.)

MySQL via a pipe is not secure, and it is done here only for the sake of clarity.

Finally, use the sample configuration file in Listing 2 as a template and edit it to your liking with a text editor:

```
cp config-sample.php config.php
vi config.php
```


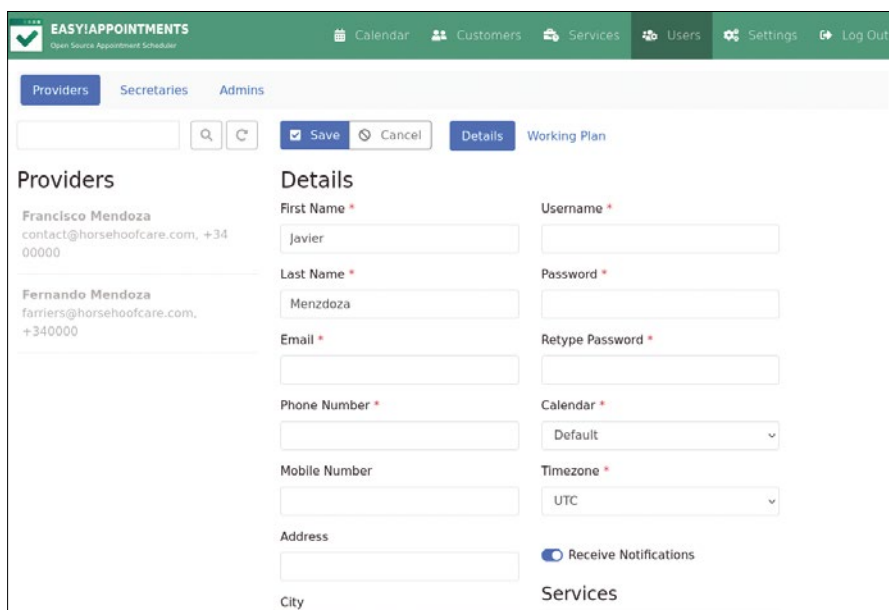**Figure 3: Add the services your business provides in the *Services* tab.**

Figure 4: Select the *Users* tab to add as many providers as desired. You can also assign specific services to each provider.

Once Easy!Appointments is loaded, you can visit your new site. An installation wizard will run automatically and help you finish the process (Figure 2).

Business Features

One of Easy!Appointments' most useful features is that it supports more than one service provider. For example, if you are setting up Easy!Appointments for the fictitious Horse Hoofcare business, which employs three farriers, you'll need to create an account for each employee. Customers then will be able to book an appointment with the farrier of their choice. To do this, go to the admin toolbar located at *yoursite.com/index.php/user/login* and log in.

First, you need to define the types of available appointments in the *Services* tab. Note that services can be grouped into categories if desired. Each service can have a price, a location, and an estimated completion time (Figure 3). An undocumented feature is that services can be defined as either *Fixed* or *Flexible* (see Table 2) [3]. In addition, you can allow multiple concurrent customers to sign up for a service, which is useful for courses and group activities that allow multiple people to sign up at the same time.

Once this is done, go to the *Users* tab and add as many providers as necessary (Figure 4). Keep in mind that you can assign different services to different providers. For example, you can assign *Infection Treatment* to a provider who is a veterinarian and *Hoof Trimming* to a farrier. If you have an employee who can provide both services, you can add both to that employee's list of services.

Finally, go to the *Business Logic* tab located under *Settings* to set your business hours as well as assign employee breaks (Figure 5).

An optional, though undocumented, step is to configure Easy!Appointments to use an email provider for delivering email notifications to both your customers and employees. Email access can be configured in `application/config/email.php` as shown in Listing 3.

Customer View

When potential customers visit your website, they will be offered a choice of services. Once the customer selects a service, a list of providers will be offered (Figure 6). After selecting a provider, the customer will be prompted to select an appointment time (Figure 7) and enter their contact information via a web form. Finally, an email notification is sent to both the customer and the employee whose services have been booked.
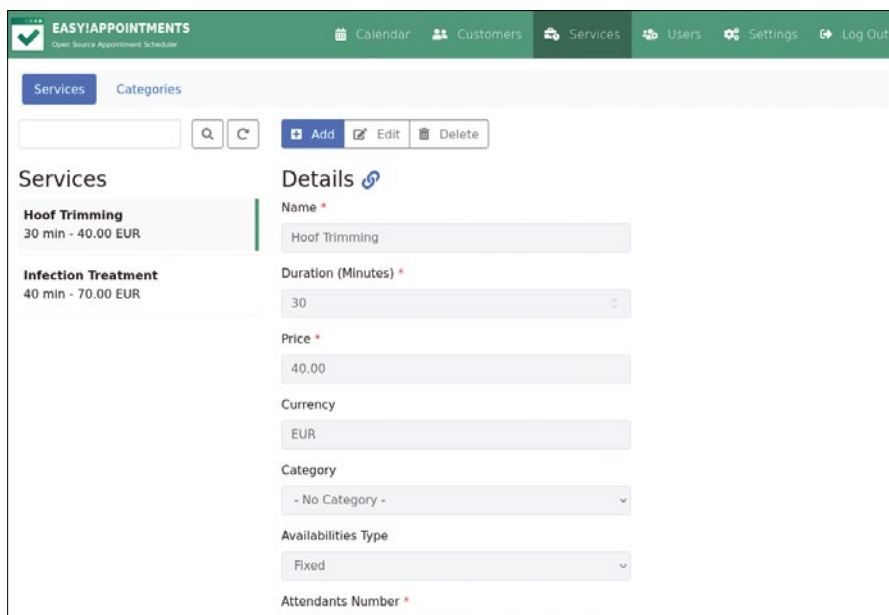
The customer can change or cancel the appointment at any time using a link included in the email notification.

Lots of Features

Easy!Appointments offers some interesting extras. First, it is a responsive web application. You will find Easy!Appointments easy to use and pleasing to the eye from either a workstation or a mobile device.

It also supports the (infamous) cookie warnings, terms of service and privacy policy pop-ups, and a very necessary CAPTCHA to distinguish humans from bots.

In addition, Easy!Appointments supports multiple languages. Both customers and employees can select their preferred language when using Easy!Appointments.

Finally, Easy!Appointments integrates with other Internet services. In particular, it features integration with WordPress [4] via a plugin and synchronizes with Google Calendar (Figure 8) [5].

At the time of writing, a GitHub project lets you create an Easy!Appointments client for Android[6], but no official releases are listed at this time.

Hidden Drawbacks

While very handy, Easy!Appointments is not free from issues. Some of the features are barely documented or not documented at all. For example, as mentioned above, Easy!Appointments can be configured to interface with an email server (in order to deliver email notifications) in a number of ways, but this is not mentioned anywhere in the README file and is not configurable via the admin toolbar. Instead, a user must edit the `application/config/email.php` file manually. While not a big deal, this could be handled better.

Easy!Appointments sorely lacks functionality when it comes to appearance customization. There is no theme engine, and it looks like the user is expected to edit the relevant CSS rules. While certainly not an obstacle for a web designer, accessing the code directly in order to customize it may not be as easy as the name implies.

Conclusion

The Easy!Appointments web appointment scheduler is easy and quick to

Figure 5: The *Business Logic* tab allows you to configure the available hours for booking appointments and to schedule breaks for your employees.

Listing 3: Example of email.php file

```php
01 <?php defined('BASEPATH') or exit('No direct script access allowed');
02
03 $config['useragent'] = 'Easy!Appointments';
04 $config['protocol'] = 'smtp'; // or 'mail'
05 $config['mailtype'] = 'html'; // or 'text'
06 $config['smtp_auth'] = FALSE; // FALSE for anonymous relays
07 $config['smtp_host'] = 'smtp.example.org';
08 $config['smtp_debug'] = FALSE;
09 $config['smtp_user'] = '';
10 $config['smtp_pass'] = '';
11 $config['smtp_crypto'] = 'tls';
12 $config['smtp_port'] = '25';
```


Figure 6: The customer selects both the service and the desired provider from drop-down lists.

deploy and includes most of the functionality any small business may need. It supports internationalization, email notifications, CAPTCHAs, and legal notices, and the technical requirements are easy to meet. On the other hand, documentation is sparse, and it appears that the lead developer's business model is to sell support to users who want site customization. ∎∎∎
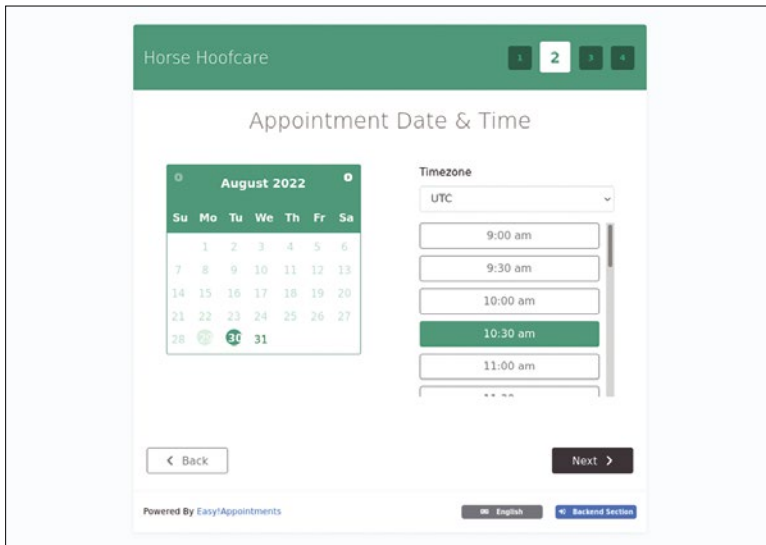
Figure 7: The customer selects a date and time from a list of available slots.

Info

[1] Easy!Appointments: *https://easyappointments.org/*

[2] Hetzner's documentation for LAMP images: *https://docs.hetzner.com/cloud/apps/list/lamp-stack/*

[3] Fixed vs. flexible appointments: *https://groups.google.com/g/easy-appointments/c/WzxsB8PLTuY*

[4] Easy!Appointments WordPress plugin: *https://wordpress.org/plugins/easy-appointments/*

[5] Easy!Appointments features: *https://github.com/alextselegidis/easyappointments#features*

[6] Easy!Appointments Android client: *https://github.com/alextselegidis/easyappointments-mobile-client*
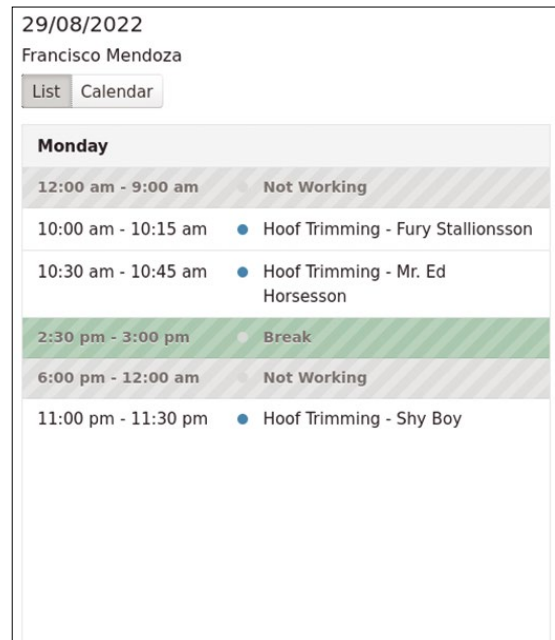
Figure 8: Employees can check upcoming appointments using the web calendar.

Author

Rubén Llorente is a mechanical engineer who ensures that the IT security measures for a small clinic are both legally compliant and safe. In addition, he is an OpenBSD enthusiast and a weapons collector.

∎∎∎

Integrating Google Authenticator
with SSH logins

Best Laid Plans

The Google Authenticator PAM module allows you to use time-based Google Authenticator passwords with various Linux services, including SSH. *By Jesse Hagewood*

I n recent years, multifactor authentication (MFA) has been a hot topic in information security, with many organizations and software services now making it a requirement. To achieve MFA, two or more authentication factors must be provided by a user to pass authentication. These factors include something you have, something you know, something you are, somewhere you are, or something you do.

Many organizations have turned to the Google Authenticator tool to implement MFA using a time-based one-time password (TOTP). Using TOTP with Google Authenticator satisfies the "something you have" authentication factor because TOTP requires a device in the user's possession (e.g., the user's Android smartphone or iPhone.) Adding a regular user password to satisfy the "something you

know" authentication factor provides the second factor to achieve MFA. Many software as a service (SaaS) providers, such as GitHub, AWS, and Microsoft Azure, support Google Authenticator as an option for MFA.

At a high level, TOTP works by having a secret key that is generated on a service and shared with a device. The TOTP algorithm with two inputs, the secret key plus the system's Unix time, results in a one-time password known by both the device and the service. A new password is typically generated every 30 or 60 seconds.

Google provides a pluggable authentication module (PAM), `google-authenticator-libpam` [1], that system administrators can use to integrate various Linux services with Google Authenticator. As a PAM module, it can be used

with virtually any Linux service with robust industry-standard authentication methods. In this article, I will specifically integrate Google Authenticator with SSH logins.

Advantages of MFA with SSH

You will find MFA useful on servers that have SSH open to the entire Internet or have SSH open to large networks. By adding MFA to SSH, you can mitigate brute-force attacks on SSH servers, as well as lower the impact of user password leaks. An attacker would need both the user's password and access to the user's Android or iPhone device (or their Google Authenticator secret key) to gain access to their user account on a server via SSH.

Typically, SSH brute-force attacks on the open Internet are dictionary

Photo by Pedro Miranda on Unsplash

password attacks, and adding TOTP will negate this type of attack. However, if a password leak is suspected, or a user password seems to have been discovered by an attacker, it is necessary to change the password regardless of whether MFA has been implemented or not.

Considerations

While MFA does provide additional security benefits for authentication with SSH, it does not make sense for all use cases. Bastion hosts, which are typically accessed manually by users or as a jump host, are usually good use cases for MFA. However, MFA may not be ideal for internal SSH hosts located behind a bastion jump host, because two separate MFA codes will be required for both the bastion jump host and the destination server. This could cause some confusion for users on login. If a bastion host configured with MFA is the only SSH server on the network with Internet access, then SSH hosts that are only accessible internally could be considered sufficiently secure with normal password or key authentication.

Large server environments primarily managed with Ansible instead of manual SSH user access pose another problem to using MFA. Ansible relies on SSH to manage configurations on servers. Having separate time-based passwords for each host SSH login during an Ansible playbook run could be cumbersome, even unfeasible, for large server environments.

Installation

The Google Authenticator module requires `libqrencode` as a dependency for generating QR codes in a shell session. On Ubuntu Server 22.04, `libqrencode` can be installed with:

```
# apt install ⤶
  libpam-google-authenticator ⤶
  libqrencode4
```

Time Synchronization

TOTP requires that the system time be accurate, so synching the server with a Network Time Protocol (NTP) service is recommended. While not a requirement for TOTP to work, NTP will ensure system time synchronization with an external service. By default, Ubuntu

Server 22.04 uses `systemd-timesyncd` for system time synchronization, but it does not have an NTP server configured by default. To determine if the NTP service is active, use the following command:

```
$ timedatectl status
```

I recommend adding NIST's NTP service as the primary NTP server, and Ubuntu's own NTP service as a fallback. To do this, uncomment the `#NTP=` and `#FallBackNTP=ntp.ubuntu.com` lines in `/etc/systemd/timesyncd.conf`, and change `NTP=` to `NTP=time.nist.gov`. After these configuration changes have been made, restart the `timesyncd` service with:

```
# restart systemd-timesyncd.service
```

The system should now synchronize its time with the NIST NTP service.

Configuration

The OpenSSH server must be configured to use PAM. In `/etc/ssh/sshd_config`, look for the configuration item `UsePAM yes`. On Ubuntu Server, `sshd` is configured to use PAM by default. Next, switch the `KbdInteractiveAuthentication` option from `No` to `Yes`.

Once these configuration changes have been added to `sshd_config`, you will need to add the Google Authentication module to PAM.

PAM looks for service-specific configuration files within `/etc/pam.d/` by default. OpenSSH server has a file located at `/etc/pam.d/sshd`. Add this configuration item to the bottom of line:

```
auth required ⤶
  pam_google_authenticator.so nullok
```

The temporary `nullok` option, used for testing, allows users that have not generated a secret key to still authenticate with SSH using only their password. Once keys have been added to all SSH users, the `nullok` option will be removed to enforce MFA.

Finally, reload the SSH server configuration with:

```
# systemctl reload sshd
```

Generating a Key

Next, you will need to generate a key for users. If you are logging in with the user account via SSH, you will see `Verification code:` after entering the password or authenticating with your SSH key. Just click Enter. Because you added `nullok` during PAM configuration, you will still be authenticated. Once logged in, run `google-authenticator` to generate a secret key. When you run the command, you will be prompted to choose whether the token is time based; enter `y` for yes.

Figure 1: For each security question prompt, enter `y`.

A QR code will be displayed, which you can scan with the Google Authenticator app on your phone. A plaintext key will also be displayed, which should be treated with the security of a password. You should write down and securely store the plaintext key or save it to a password manager. This plaintext key can be used to re-add the TOTP key to Google Authenticator if the existing phone is lost, replaced, or broken. After adding the QR code or key to your Google Authenticator app, you can enter the current code displayed in the app for the next prompt in the shell console to verify it is working (you can use `-1` to skip this, but it is not recommended). You will also receive several "scratch codes," which can be used in emergencies if the Google Authenticator device is not available. Next, you'll be given several prompts for security options; choose y for yes for all of them (see Figure 1) The rate limiting options are especially important because these options will prevent an attacker from attempting a brute-force attack on a time-based password.

Finalizing

Once a secret key has been generated for every user that will need to use SSH, you must remove the `nullok` option to enforce MFA. As a critical step before continuing, you must test the `google-authenticator` module to make sure it is working on your system, as well as confirm that all existing users have set up secret keys and Google Authenticator on their devices. Once this step is complete, in `/etc/pam.d/sshd`, change:

Listing 1: Generating a Temporary Key

```
# su - new_user
$ google-authenticator
$ exit
```

```
auth required ⏎
  pam_google_authenticator.so nullok
```

to:

```
auth required ⏎
  pam_google_authenticator.so
```

Then reload `sshd`:

```
# systemctl reload sshd
```

Adding New Users

If you periodically add new users to the server that require SSH, these users will need to generate a key via a direct console login. If a console login is not feasible (e.g., servers running in a cloud service), then a temporary key will need to be generated by the system administrator or ideally by a user creation script. An administrator can generate a temporary key for a new user with the code in Listing 1.

Once generated, you then share the temporary secret key, QR code, or plaintext key with the user. The new user will ideally need to rerun key generation on their first login, which you can enforce with various methods such as using a flag file for new users.

Flag File Enforcement

The `/etc/skel` directory contains default shell profiles and configurations that are copied into users' home directories on account creation. You will need to update `/etc/skel/.profile` to check for a flag file. Flag files are empty files that are often used in shell scripts to determine how the script should behave. Append the `if` block in Listing 2 to `/etc/skel/.profile`.

The modification from Listing 2 will check if the flag file `$HOME/.first_login` exists. If so, it will run

`google-authenticator` and then delete the flag file. When new users are added to the server by the administrator, the administrator can manually create the flag file in the new user's home directory with the commands in Listing 3.

To optionally automate this, you could create an empty flag file inside of `/etc/skel`:

```
# touch /etc/skel/.first_login
```

By having the flag file inside of `/etc/skel`, the flag file will be placed in the new user's home directory automatically, so the administrator will not need to `touch` the flag file after creating the account.

Conclusion

With simple setup and low administrative overhead, adding Google Authentication TOTP as an authentication method for SSH provides the additional security of MFA for remote server access at a low time cost. ∎∎∎

Info

[1] google-authenticator-libpam: *https://github.com/google/ google-authenticator-libpam*

Author

Jesse Hagewood is a Certified Information Systems Security Professional (CISSP) and an AWS Certified DevOps Professional. He has been a Linux administrator since the start of his career, beginning with managing self-hosted Linux email and web servers for local newspapers and later on administering large SaaS platforms. Currently, he is a DevOps Engineer at Ozmo, Inc. in Blacksburg, Virginia

Listing 2: Modify /etc/skel/.profile

```
# Run google-authenticator if a flag file exists.
if [ -f $HOME/.first_login ]
then
  google-authenticator
  rm -f $HOME/.first_login
fi
```

Listing 3: Adding a New User and Creating a Flag

```
# useradd -m new_user
# passwd new_user
# su - new_user
$ google-authenticator
$ exit
# touch /home/new_user/.first_login
```
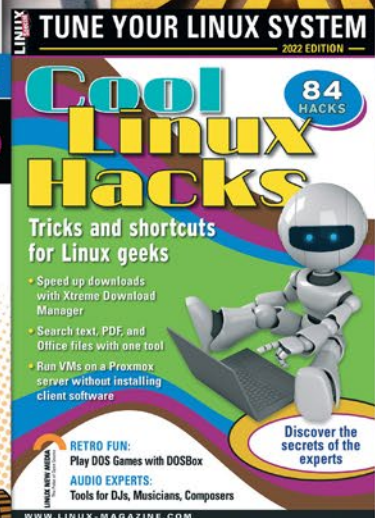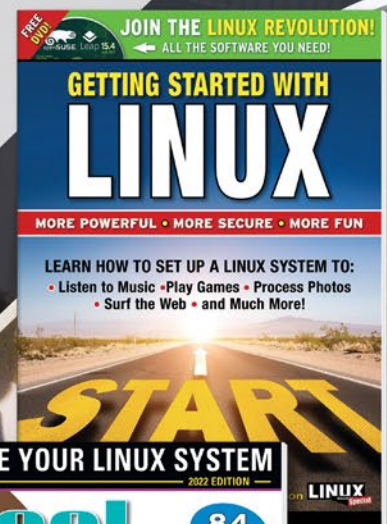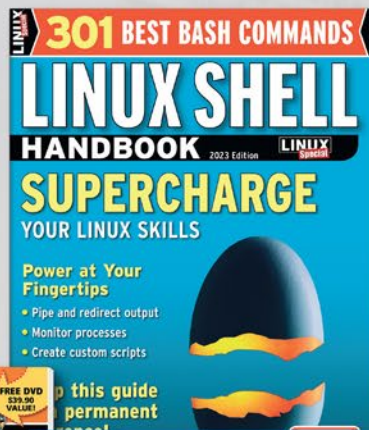
MakerSpace

Free MathCAD alternative for Linux

Computing Competency

Present complex equations with intermediate steps, graphics, plots, and results in SMath. *By Brooke Metcalfe*

U niversity students in engineering often have assignments in which they must show their work step-by-step with sample calculations. Although Excel and Python are useful for advanced calculations, they can't present complex equations formally in a report.

MathCAD [1] is a good package for writing engineering reports. The tool offers a workbook interface that allows you to enter complex equations in a readable format, along with a rich function library, programming interface, graphics, and plots. Unfortunately, MathCAD isn't supported in
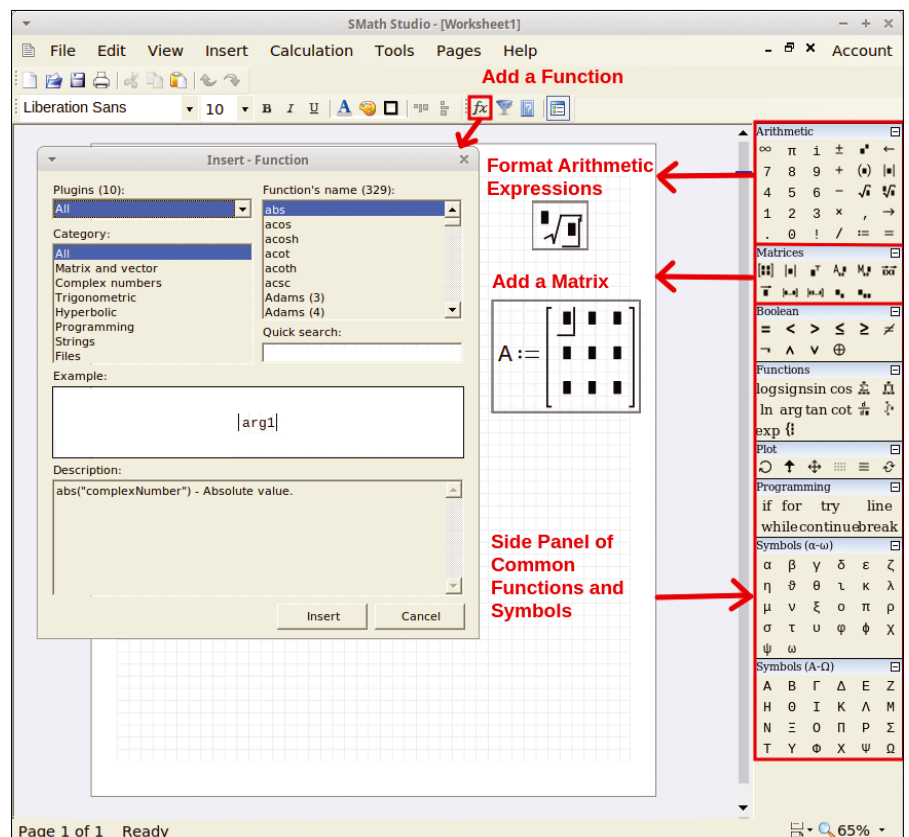
Figure 1: Panel of common functions.

Linux, and it is generally out of the budget for students or casual users. As an alternative, SMath is a great free

Figure 2: Editing and viewing a calculation.
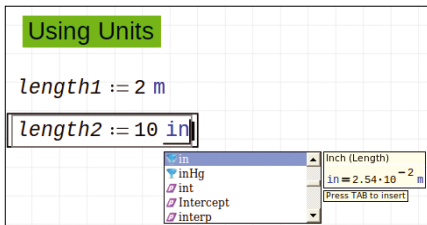
Figure 3: Inserting units.

option that works in Linux, macOS, and Windows.

In this article I introduce SMath and show you some of the useful features that I expect to use in my next engineering term. I also look at an SMath example that solves a typical high school or first-year university math problem.

Getting Started

The SMath installation files reside on the project's website [2]. To install the software, the Mono development environment needs to be loaded first. In Debian/Raspbian/Ubuntu systems, enter:

```
sudo apt install mono-devel
```

In the next step, copy the SMath installation files to a local directory on your laptop and run the SMath desktop:

```
# run SMath Studio in Linux:
./smathstudio_desktop_mono
```

SMath Studio has a side panel of common functions and symbols (Figure 1). All the functions are listed in the Insert Function (*fx*) dialog box.

Equations, Variables, and Units

SMath lets you create complex calculations from the side panel (Figure 1) or with keyboard shortcuts. Complex equations can be modified with the arrow keys to move the cursor between elements (Figure 2).

SMath Sheets calculate from top left to bottom right, so it is important to define a variable before it is used. A variable is defined with **:=**, and the results of a calculated value are displayed with the equals sign (**=**).

To add units to a variable, type a single quote (**'**) before entering the unit, or you can choose from a list of units (Figure 3). If a calculation has mixed units, SMath automatically manages the conversion between units (Figure 4).

Figure 4: Mixing units.

Figure 6: Defining variables with interface widgets.

Figure 5: Inserting a slider.
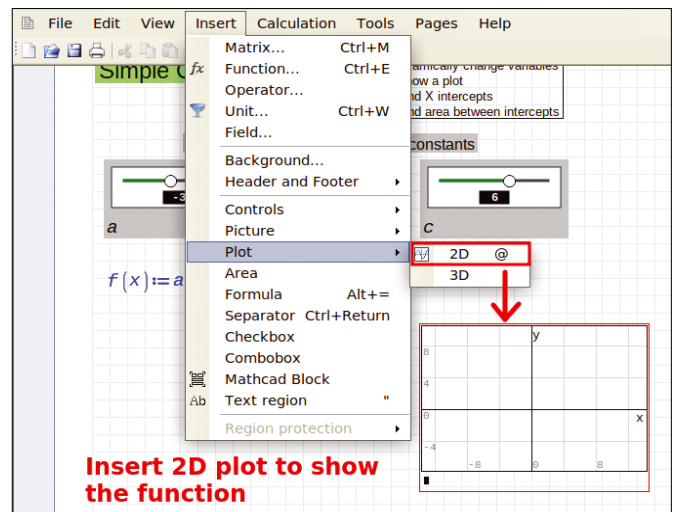
Figure 7: Inserting a plot.

To make things a little more presentable, SMath shows variables, calculations, and results in black and units in blue, although this feature is adjustable. The final results of a calculation can be presented in fractions, unit conversions, or simply the numeric value.

Solutions

SMath also supports useful features such as interface widgets and plotting, as well as conditional code. One problem that regularly needs to be solved in high school and many first-year university courses is to find x intercepts and the area under a parabola defined by a quadratic equation. The next example solves this problem and highlights some of the key features built into SMath.

The first step in solving this problem is to define the input variables. User interfaces such as adjustable sliders can be added from the Insert menu in the toolbar (Figure 5). *Slider* widgets define and toggle variables (Figure 6).

Plotting a parabola is a quick way to check the number of x intercepts for an equation. To add a plot, use *Insert | Plot | 2D* (Figure 7). Once a plot is dropped onto a workbook it can be resized and repositioned. For this

example, the plot is configured to show the $f(x)$ quadratic equation.

You can quantitatively find the number of x intercepts in two ways (Figure 8): by manually solving the quadratic formula for x when $y = 0$, or with the SMath `solve(2)` function. To insert this function, type *solve*, then press the Tab key once the dropdown menu displays *solve(2)*. To find the area under the curve, the results of the `solve` function should be defined as a new variable (e.g., `MyRoots`).

The `solve(2)` function returns one or two x intercepts where the curve touches or crosses the x-axis. If you don't change your settings to *Calculation | On error | Continue* (Figure 9), the logic will be interrupted if a curve does not cross the x-axis.

Added logic can handle special cases. To solve for the area under the curve, I used the `if` statement then checked for valid roots (Figure 10). An `IsDefined` function

checks whether the *MyRoots* variable is valid (i.e., has roots).

If roots are found, the limits are passed to an integral function to calculate the area. (Integral and derivative functions are found in the *Functions* side panel.) If no roots are found, the string `"Equation doesn't cross axis"` is passed to the `area` variable.

Figure 11 shows the final SMath worksheet example, with the interface widgets and plotting and conditional code that find the x intercepts and area under a parabola with the quadratic equation.

Examples

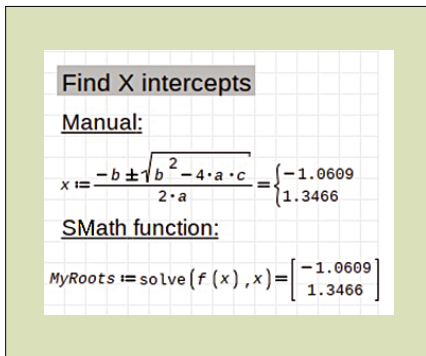The package comes with a variety of examples of modifiable calculations. One
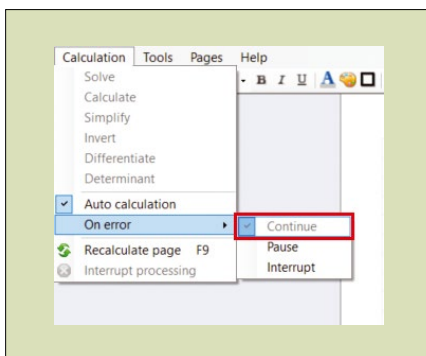
Figure 10: How to insert conditional code.

Figure 8: Finding *x* intercepts.

Figure 9: Changing calculation settings.
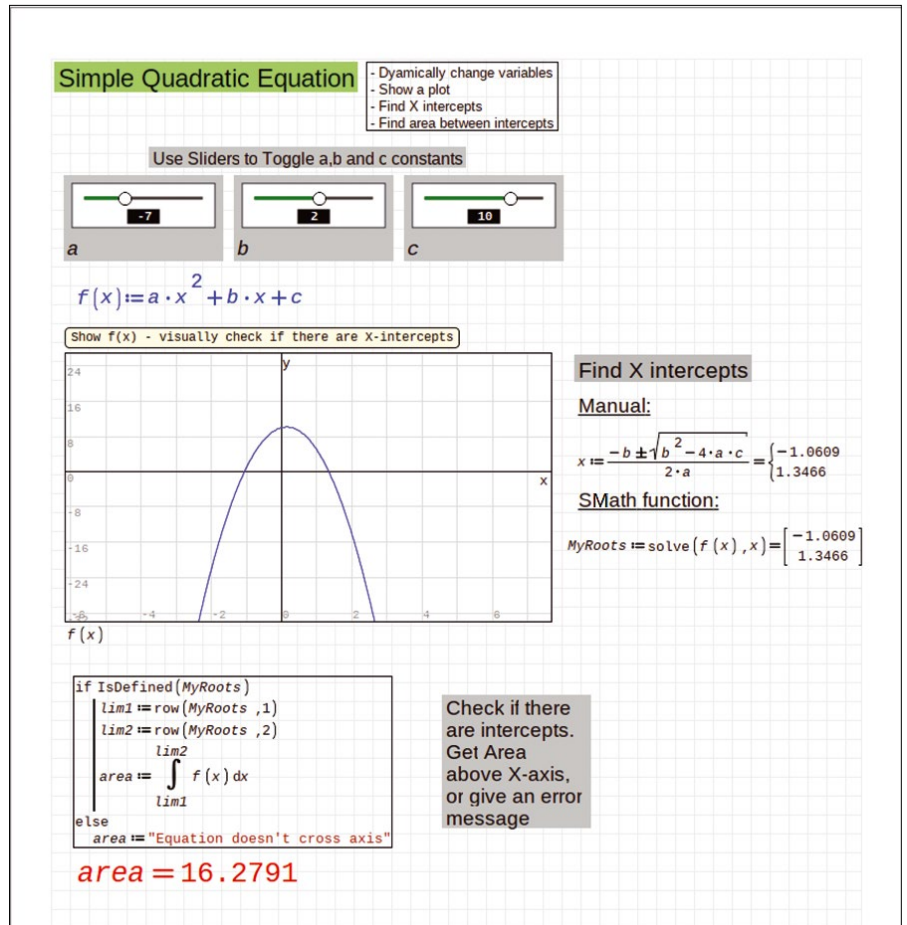
Figure 11: Slider, plot, and code example.

Figure 12: Example beam calculation.

of the examples is a beam support reaction (Figure 12), which displays a diagram of a beam that automatically adjusts with given variables. I was able to make use of this example to help clarify some questions in one of my civil engineering classes.

Summary

Spreadsheet packages such as Excel are great tools, but they are weak at clearly presenting complex equations. SMath, on the other hand, presents equations in their conventional forms and has worksheets with customizable layouts.

As a university student, I spend a lot of time formatting lab reports. In the past, I used static tools such as LaTex [3] or the Microsoft Word Equation Editor to build sample calculations. Now I find that the dynamic presentation of complex calculations in SMath can be a huge time saver. ∎∎∎

Info

[1] Mathcad: *https://mathcad.com*

[2] SMath: *https://en.smath.com*

[3] LaTex: *https://www.latex-project.org/*

Author

Brooke Metcalfe is in her third year of Environmental Engineering at Carleton University in Ottawa, Canada.

MakerSpace

Four Raspberry Pi advertising and tracking blockers

Trustworthy

A Raspberry Pi with the right software filters out annoying ads and nasty trackers for end devices on your local network.

By Erik Bärwaldt

Author

Erik Bärwaldt is a self-employed IT admin and technical author living in the United Kingdom. He writes for several IT magazines.

Advertising on the Internet can be intrusive and annoying, often with trackers that spy on web browsing behavior. For standalone workstations, such unwanted content can be easily restricted or blocked by browser extensions. However, if you want to configure several workstations, setting up the extensions can take a great deal of time. Luckily, special appliances can block unwanted content centrally before it reaches the intranet. I looked at different strategies and their implementations for blocking ads and trackers on websites.

Functionality

The solutions presented in this article can be integrated directly downstream of the router on the local network so that all incoming and outgoing data traffic runs through the tools. These appliances act as DNS servers in this process, except upribox, which sets up a wireless network you can use to access the Internet securely. Some of the solutions also offer integrated VPN servers and anonymize IP addresses. From outside the VPN, the workstations on the internal network can no longer be identified by their IP addresses. All of the solutions discussed are also free software, and the hardware usually is based on a Raspberry Pi.

One of the main advantages of centralized appliances is that you only need to connect and configure them once on the local network. For the most part, the systems automatically update during operation, which avoids additional overhead for customization and regular updates. Manual configuration of the clients is also largely eliminated. The tools support a wide variety of devices. In addition to computers, smartphones, tablets, and Internet of Things (IoT) devices are automatically protected.

AdGuard Home

The AdGuard Home [1] DNS server blocks certain domains that distribute advertising and spy on surfing behavior. Known malware domains can be blocked with blocking lists. The software does not require a client on the end device: Just install it on a Raspberry Pi and it protects the whole network.

As its home base, AdGuard requires an installed operating system, such as Raspberry Pi OS. The recommendation is to give the small-board computer (SBC) a static IP before the install to avoid problems caused by an address change after a reboot. The application can then be installed and started with a single command at the Raspberry Pi's terminal:

```
$ curl -s -S -L ⮑
  https://raw.githubusercontent.com/⮑
  AdGuardTeam/AdGuardHome/master/⮑
  scripts/install.sh | sh -s -- -v
```

Lead Image © Vitaliy Gaydukov, 123RF.com

The software automatically detects the hardware you are using, sets up the system, and shows at the end of the routine information on how to access the tool's web interface and control the application at the prompt. As soon as you call the specified URL in a web browser, a graphical setup wizard launches. The wizard guides you through the basic configuration in five steps, requiring you to create a user account and password to protect the system.

The wizard also provides detailed instructions for configuring the router and various device classes. For the AdGuard home server to work correctly, you need to modify the DHCP and DNS settings on the router for Internet access; otherwise, problems could occur. After completing the basic setup, log in to the web browser and access the dashboard (Figure 1).

The dashboard displays statistics about the connected clients, DNS requests, denied requests, and filtered web pages in a graphical form. Underneath is a table of clients, a list of the most frequently requested domains, and a list of the most frequently blocked domains. AdGuard Home does not update these statistics continuously, but you can update them manually by clicking *Refresh statistics*.

After closing and reopening the administration interface, you do not need to specify the port number in the URL unless you changed the default values during the basic setup.

Various services (e.g., a parental control filter) are enabled in the *Settings | General Settings* menu. In this dialog, you also specify how often you want AdGuard Home to update the blocking lists. The *Enforce safe search* option blocks predefined web pages.

Depending on the size of the local network and the volume of the data traffic, you might want to modify the log intervals. In *General settings | Log configuration*, you can define how long the server keeps the logs and whether the client IP addresses are anonymized.

In *Settings | DNS settings*, you can enter additional DNS services on top of the existing upstream DNS server. With multiple DNS services, the software supports load balancing and gives priority to the fastest server, but you can change this behavior by clicking on the

radio button to the left of one of the alternatives.

If the logging function is enabled, AdGuard Home lists all requests in the *Query log* tab. It not only lists the times, the requested servers, and the status messages of the requests, but also – in the *Client* column – the computer on the local network from which the request originated. You can use these logs to readjust the filter settings if necessary. For individual clients, certain server requests can be blocked by clicking *Block* (Figure 2).

The *Filters | DNS blocklists* menu already contains two filter lists with a total of 55,000 blocked domains. Checking a box enables a list, whereas clicking the *Check for updates* button lets you update the enabled lists.

To add additional lists, click *Add blocklist*. In the next dialog, you can then decide whether or not to enable one or more items from a list of predefined blocking lists (Figure 3) or to add your own blocking list. The selection dialog for predefined blocking lists
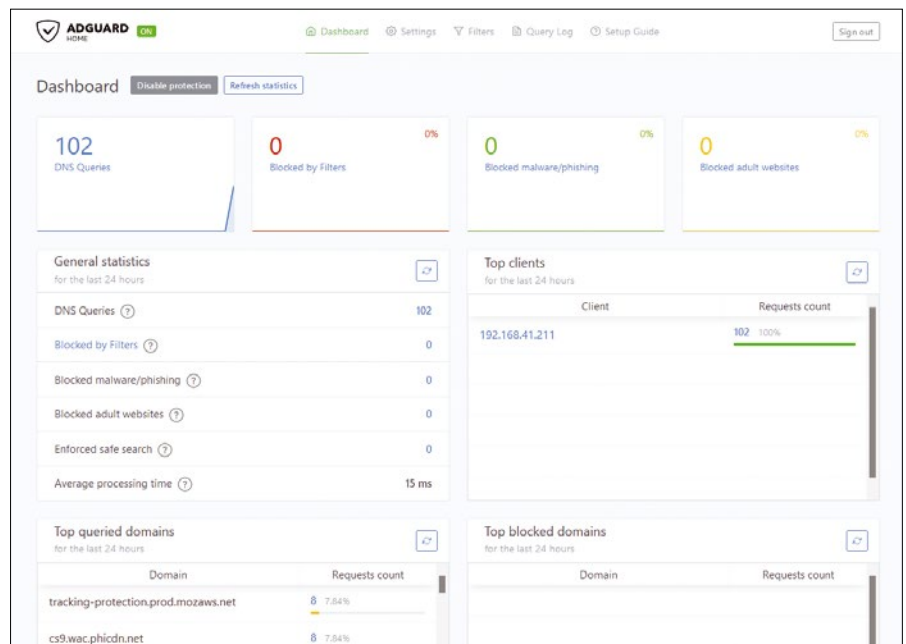

Figure 1: AdGuard Home's start-up window is straightforward.
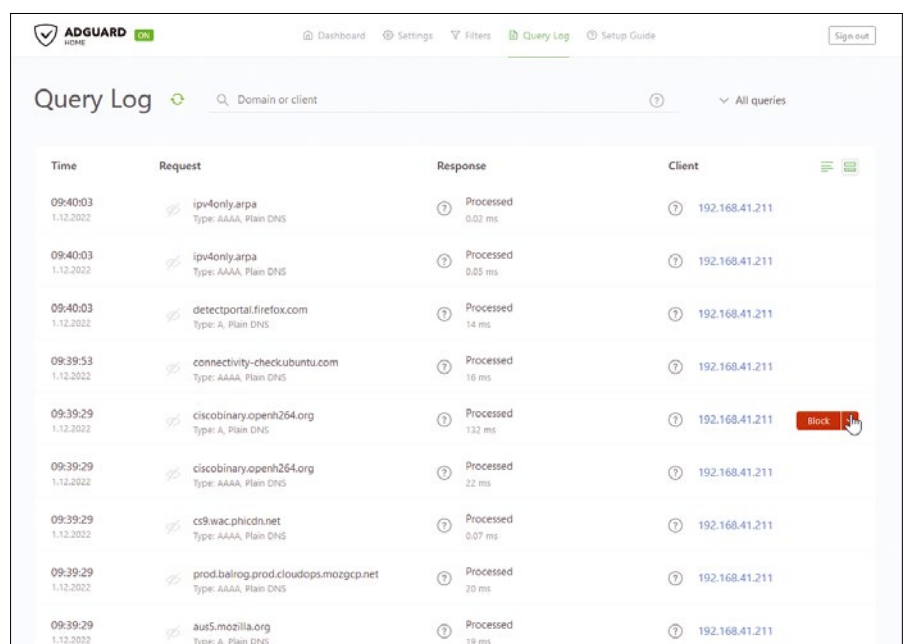

Figure 2: The AdGuard Home logging function supports manual domain blocking.

lets you include general lists with advertising and tracker domains and integrate malware domain lists. Also, you will find regional lists sorted by country.

For heterogeneous environments with Windows clients, you can also enable a spy list specially adapted to the vulnerabilities of your operating system.

eBlocker

eBlocker [2] is a complete solution for the Raspberry Pi, the Banana Pi M2 + , and VirtualBox virtual machines. eBlocker started life as a hybrid hardware and software solution. However,

the manufacturer stopped distributing commercial eBlocker devices in 2019, while continuing to maintain the software as eBlockerOS in what has become a non-profit project.

Developers of eBlockerOS stipulate a minimum requirement of a Raspberry Pi with at least 1GB of RAM, but they recommend a fourth generation model with at least 2GB of RAM. The system requires an 8GB SD card as a minimum that should at least meet the recommended class 10 specification [3]. In terms of the power supply, the project advises you go for at least 3A. Additionally, you need a free local area network (LAN) port on the router or switch to connect the Raspberry Pi.

The developers also provide a list of compatible routers and repeaters that have been tested in combination with eBlockerOS. The list does not claim to be complete, but most commercially available devices and the all listed browsers are compatible, although some

browsers have certain functions that cannot be used in combination with eBlockerOS.

To begin, download the latest eBlockerOS image, which weighs in at nearly 740MB, and unpack the image with an archiving tool. The resulting image is about 4.2GB, which you can transfer to a microSD card with a tool such as balenaEtcher [4] or the command:

```
$ dd ⤶
  if=eBlocker-2.8.2-raspberry-pi.img ⤶
  of=/dev/mmcblk0 bs=4M
```

From the image, boot the Raspberry Pi, which must be connected to the router with a standard LAN cable. The Raspberry Pi does not need a keyboard, mouse, or screen.

After a wait of a couple of minutes, you can enable eBlockerOS on the Raspberry Pi with any workstation on the network from *http://setup.eblocker.org*. The eBlocker system now uses address resolution protocol (ARP) spoofing to locate end devices on the LAN automatically and to intercept the data traffic as a gateway. A small orange eBlocker icon appears in the upper right corner of the screen, along with some status information relating to the current system. From this point on, eBlocker analyzes all data packets with the help of deep packet inspection, additionally checking notorious data collector domains by DNS blocking and modifying the user agent identifier that is sent.
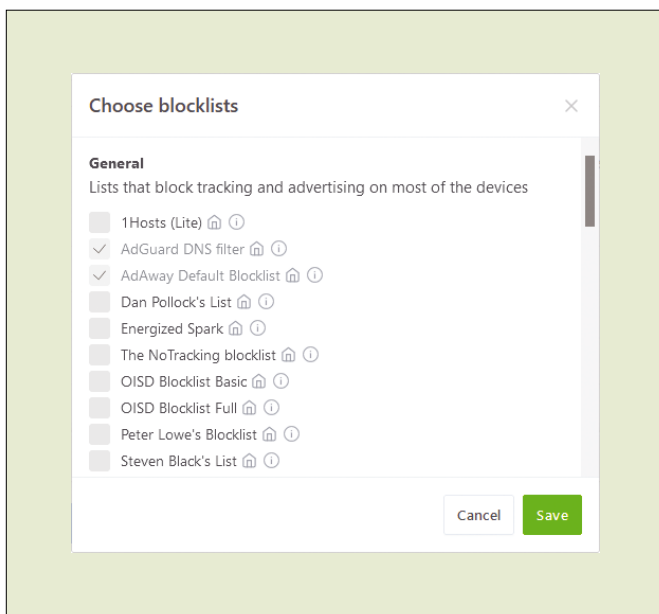

Figure 3: In AdGuard Home, you can add additional blacklists at the push of a button.
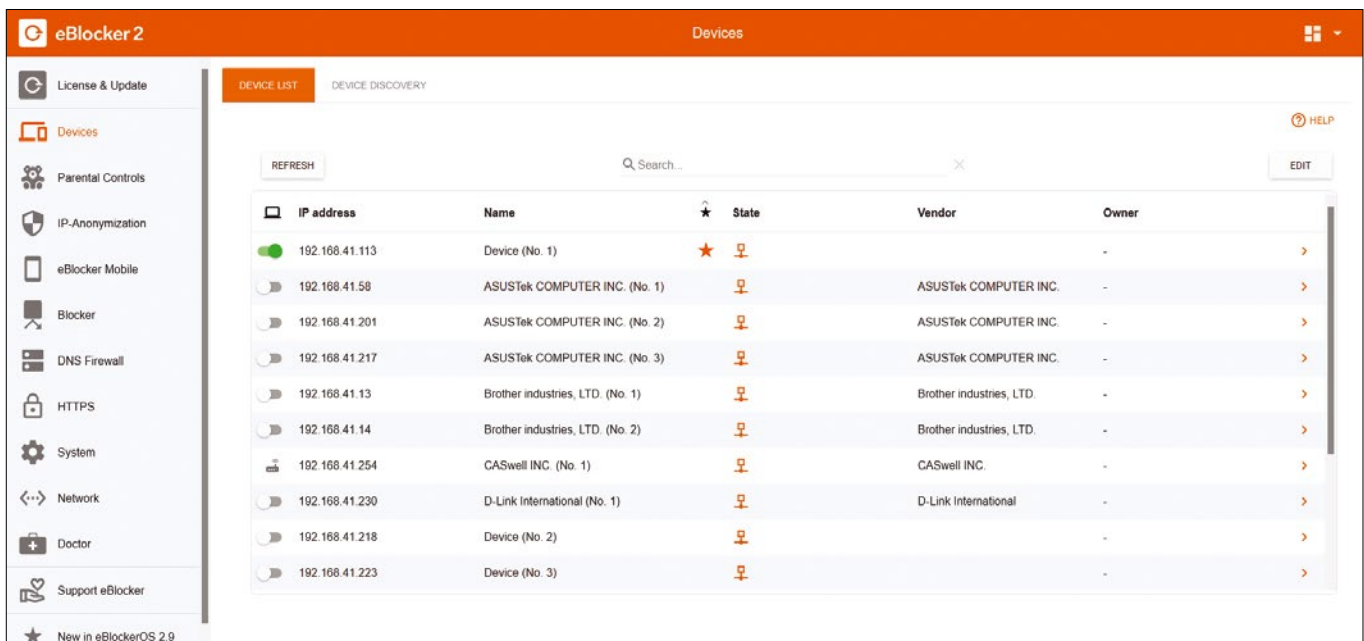

Figure 4: The eBlocker configuration options are grouped sensibly.

To configure eBlockerOS, click on *Dashboard for this device*. A dialog appears telling you that you can create a bookmark for the browser. You are then taken to a spartan dashboard. Click on the gear icon in the top right corner of the dashboard to configure the system. In the setup wizard, select *Continue*, accept the license agreement, and then go on to set the time zone in the next step. The wizard prompts you for a name for the eBlocker system, but this name is only important if you will be running multiple eBlockers on the LAN.

The next step relates to automated activation of eBlocker for devices that you add to the network. By default, eBlocker is not enabled for new devices. In the last step you need to enter the license key, a relic from the time of the commercial eBlocker version. A valid license key will already be displayed, and you just need to enter a valid email address. This address is used to contact you at short notice in case of emergencies.

If updates are available, the wizard prompts you to update eBlockerOS. After completing the update, the Raspberry Pi reboots, and a notification window appears telling you about the installed updates.

After clicking the *Continue* button again, you will be taken to the dashboard (Figure 4). In the *Device List*, you will see all the devices on the LAN that the system located during setup. From the settings bar, you can configure the various options as needed. In the future, you will be able to access the dashboard by entering the IP address of the eBlocker system followed by port number 3000.

To configure the system in detail and protect it from unauthorized access, first set an admin password by clicking *System* at bottom left in the vertical bar and then the *Admin Password* link.

You can also change other security options. To anonymize the IP addresses, eBlocker provides two options in the *IP Anonymization* group: By default, data packets are routed through the Tor network, although a VPN connection with a public service provider can be configured as an alternative.

You can also set various options for the DNS firewall, parental controls with blacklists and whitelists, and mobile device protection in the appropriate categories. You can also view status indicators in the *Doctor* category. The *Devices* dialog box is where you switch eBlocker protection on or off for individual terminal devices.

For each terminal device on the LAN, you can view up-to-date status information in the web browser. Type the IP address of the eBlocker system in the browser, followed by a colon and port number 3000. Alternatively, just type *http://eblocker.box*.

eBlockerOS opens a window with numerous status "cards" (Figure 5). You can view system messages in the *Messages* card or enable IP anonymization for the current workstation from the *Anonymization* card. You can also check the function of the tool or pause eBlocker for the workstation from the card at top center. Also useful is the option to manage lists of domains you want to block or allow by adding them to the *Block Domains* and *Allow Domains* cards.

Pi-hole

Pi-hole [5], designed for the Raspberry Pi but also available for Intel/AMD architectures, is one of the best known ad and tracking blockers. The appliance requires at least 2GB of free disk space and 512MB of RAM. Pi-hole turns the Raspberry Pi into a DNS server that filters all data packets. You can even save profiles for defined terminal devices (e.g., to block adult websites).

You can add the Pi-hole application to an existing operating system with `curl`. Before you install the application on Pi OS or Debian, the best approach is to enable an SSH server so that you can later access the Raspberry Pi from any terminal on the network. On Pi OS, you simply enable the SSH server with a slider in the operating system's configuration dialog. On Debian, you need to install the *openssh-server* package manually and install Pi-hole in a terminal window with:

```
curl -sSL ⤵
    https://install.pi-hole.net | bash
```
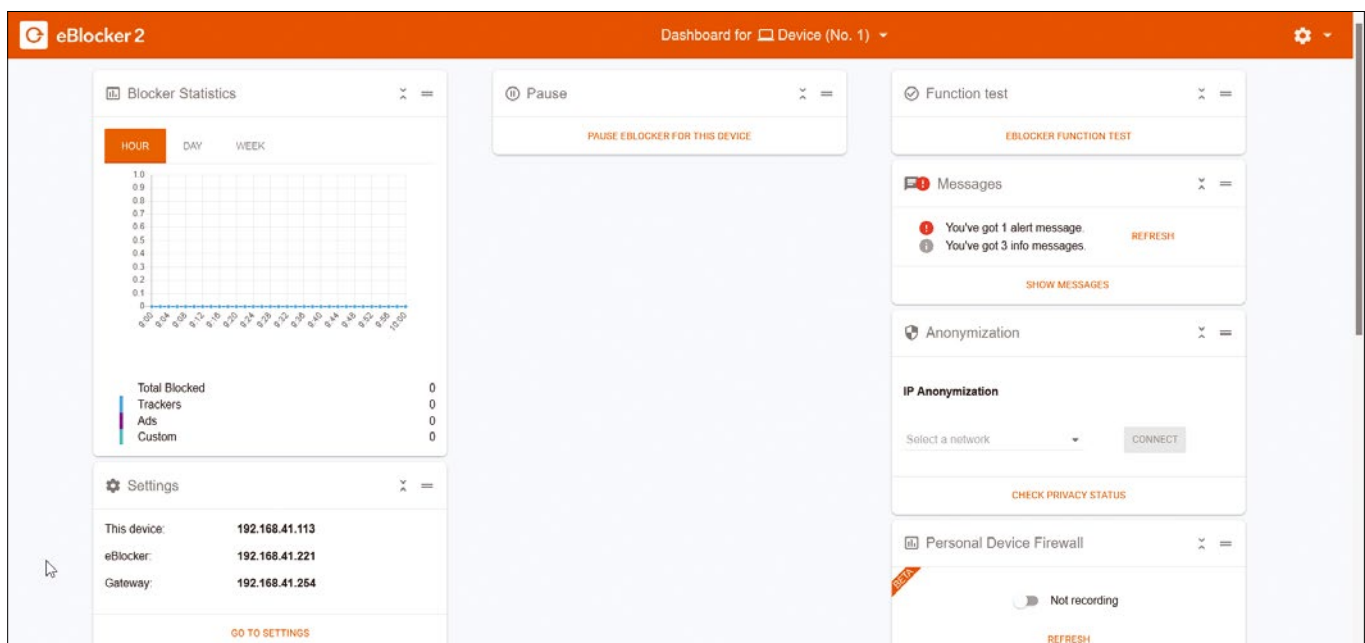

Figure 5: Statistics displays in eBlocker give an overview of filtered content.

After the install, an ncurses configuration wizard pops up automatically.

In the wizard, first set a static IP address for the computer, select a DNS provider from the list (Pi-hole uses this provider to resolve web requests), and load a block list. The next step is to enable the web interface for management and specify some more details for logging the data. The software then displays a screen with all the important data, including the password for logging in to the web interface. A message tells you that you need to enter the Pi-hole computer as the DNS server on all the terminal devices on your LAN.

You can now access the Pi-hole system's web interface from any web browser on the local network. Type *http://pi.hole/admin* or the Pi-hole IP address. After logging in, you are taken to an interface that groups the settings options for configuring the tool (Figure 6).

You will find the most important data relating to existing clients, their requests, blocked calls, and filter lists displayed in a graphically appealing way. Below this information, bar graphs visualize the number of network requests, followed by pie charts and tables on the most frequently accessed domains, which

Pi-hole sorts by allowed and blocked domains and the type of request. The system also shows as percentages which upstream servers were used. At the very bottom of the window is additional statistical information about the clients logged in to the network.

One key feature in Pi-hole is its ability to group individual endpoints and apply separate filtering rules to each. For example, you can group tablet PCs or smartphones that are predominantly used by children in a separate group and define uniform rules for the group. This filtering option eliminates the time-consuming configuration of individual devices.

In the *Groups* menu in the sidebar, you can create arbitrary groups, which you then enable or disable individually with sliders. Just below *Groups* is the *Clients* option, where you create individual clients. The terminal devices discovered by Pi-hole appear here along with their MAC addresses in a drop-down menu. You can create a separate entry for each client on the LAN and assign each of the clients to a group, as shown in a table.

One of two approaches can be taken to extend Pi-hole's blocking lists: To

add a single domain to the blocking list, you would use the *Domains* dialog. You can also opt to include all subdomains in the list. Alternatively, you can add lists of domains to be blocked in the *Adlists* section by typing the URL of the new list in the address field and then running the `pihole -g` command in the terminal to add the listing to the system. Alternatively, you can follow the link provided in the dialog to update the existing block lists in the browser window.

The newly added lists appear in a table below the input dialog and can be removed again, if needed, by clicking on the red trash can icon on the right of the list entry. By default, Pi-hole enables new blocking lists for all groups. However, you can also assign new lists specifically to individual groups of terminal devices. Newly added lists can be switched on or off at any time during operation with the slider in the *Status* column.

To disable Pi-hole for specified periods of time, select the appropriate option from *Disable Blocking* in the sidebar by specifying periods during which you do not want the application to run any blocking functions.
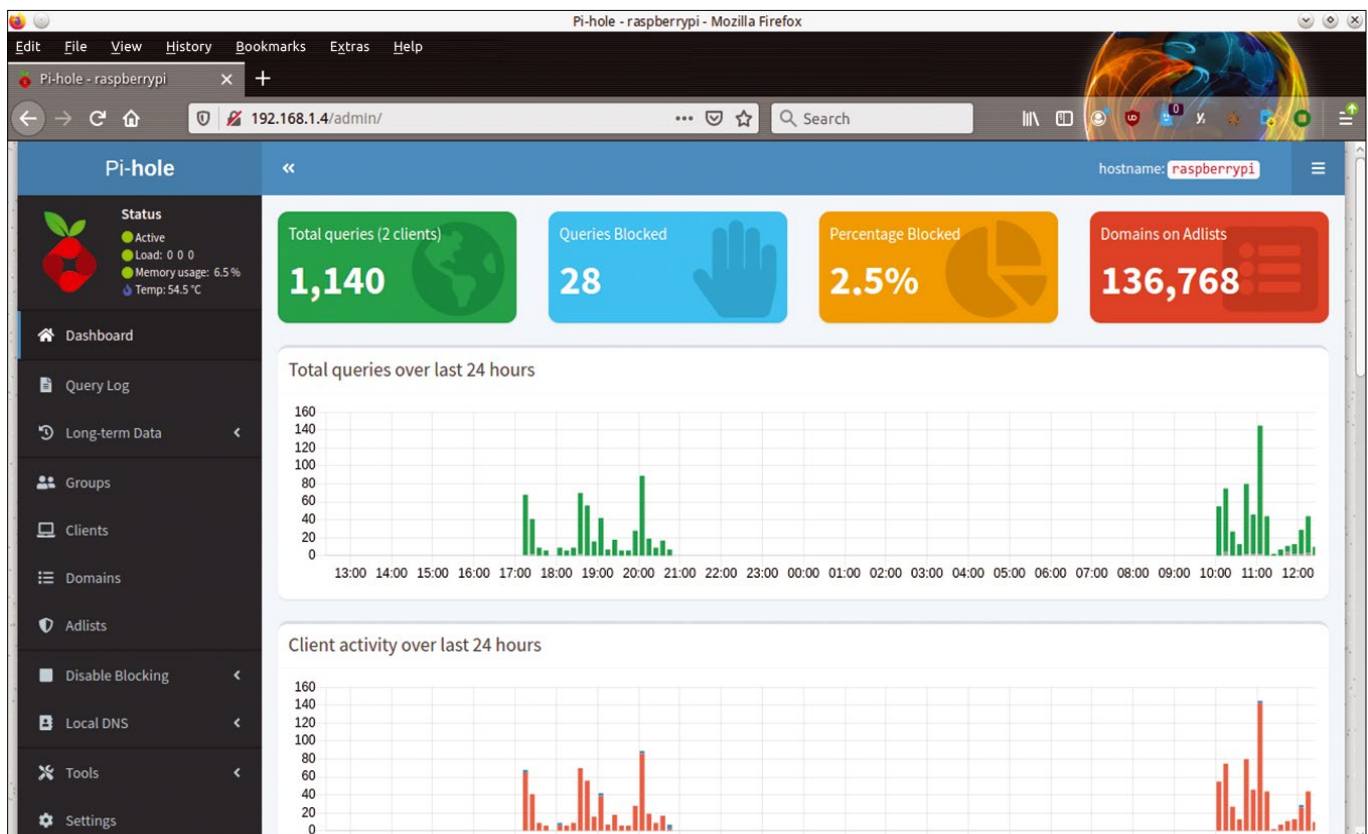

Figure 6: The Pi-hole dashboard has an ergonomic and contemporary design.
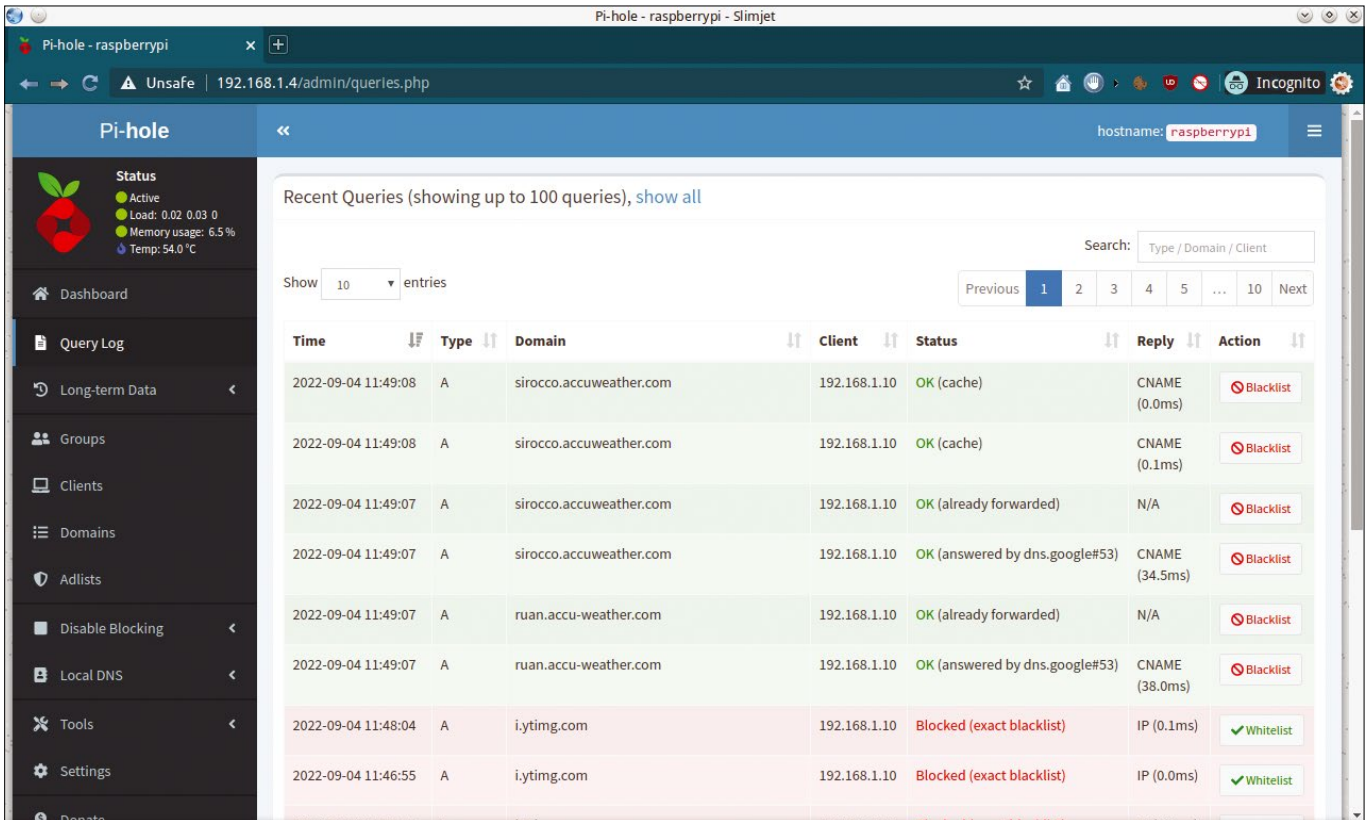
Figure 7: Selectively add individual domains to blocking lists in Pi-hole.

One hundred entries are shown in each case. If you want to add one of the domains to a blacklist, click *Blacklist*. After confirming the prompt, Pi-hole adds the domain into the additional list of individual domains to be blocked (Figure 7).

upribox

Upribox is one of the senior citizens among central filtering solutions for intranets. You can download the free software for your Raspberry Pi (older third generation only) as a 655MB image from the project's GitHub page [6].

The upribox already sets up a WiFi network at initial startup, but you do need to connect to the LAN over a cable connection to complete the setup. Working on any computer on the network, log in to the *upribox* WiFi network with the *changeme* WPA2 key and call the URL *https://upri.box:4300* in the machine's web browser. After a security prompt, you are taken to the system's administration interface. On the login screen, log in with the *upri* account and *changethedefaults!* password. You are now taken to the dashboard with the device overview (Figure 8).

To begin, change the password for the *upri* user by clicking on the username top right in the browser window. The

software now opens a dialog where you can enter a new password and change the name of the administrative user.

For each device logged in to the wireless LAN, you will also find its status in

the dashboard. Upribox lets you change the protection status of each workstation on the WiFi network individually by setting a radio button. The devices are in silent mode by default, which hides ads
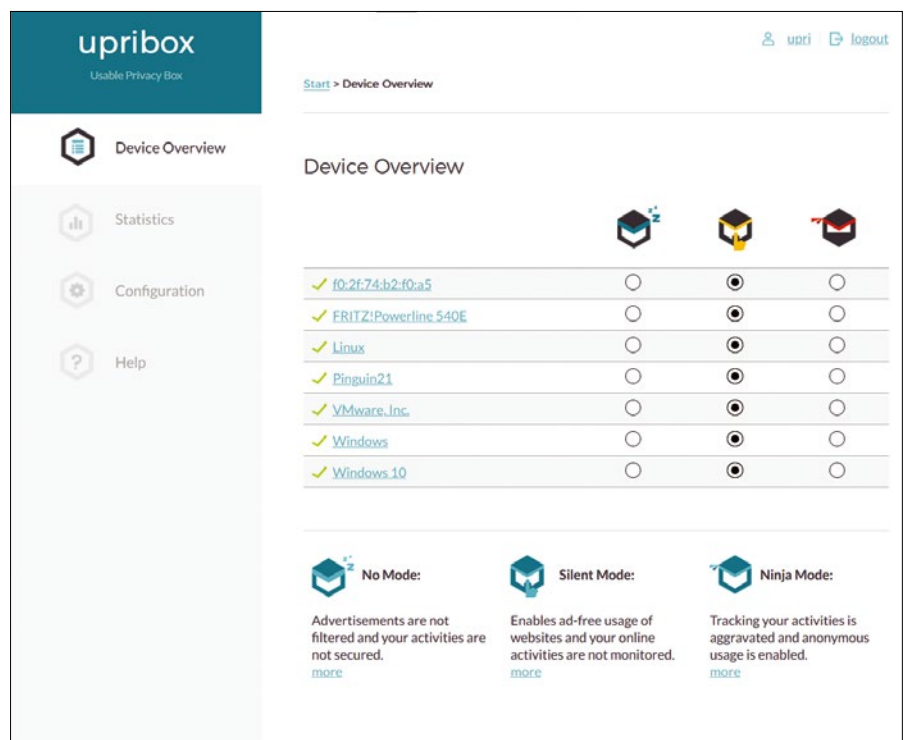
Figure 8: The upribox dashboard looks simple, but it gives you all the information you need.
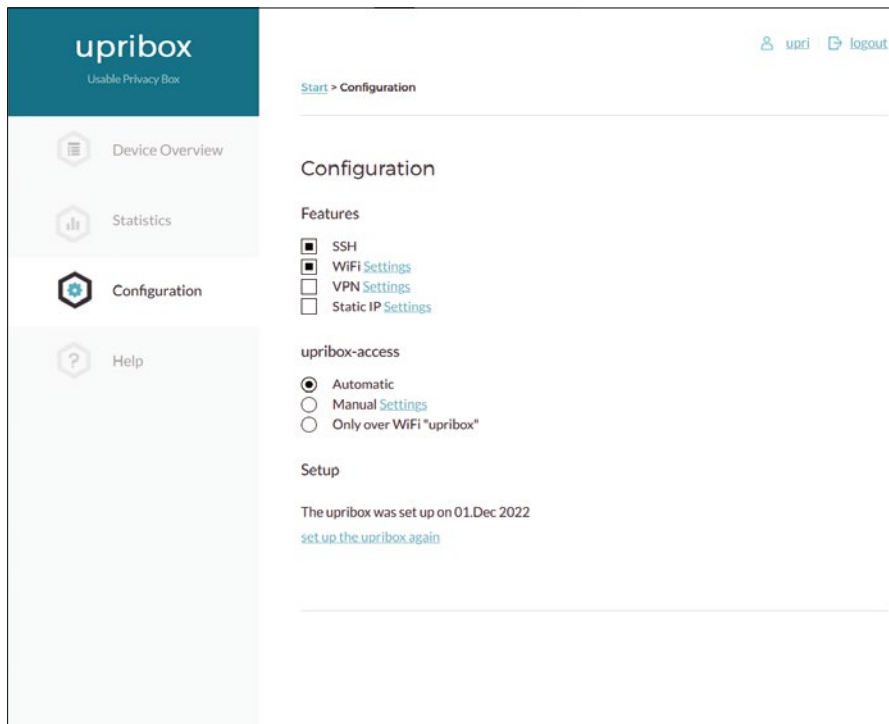
Figure 9: The upribox configuration dialog is fairly spartan.

centrally filtering advertising and trackers. The differences lie in the feature set and user ergonomics (Table 1). The easiest solution in terms of handling is upribox, which offers only a few settings.

Pi-hole and eBlocker have the largest feature sets. Both require some basic knowledge of network technology. Pi-hole impresses right from the outset with the most extensive blocking lists. Like upribox, AdGuard Home is primarily intended for use on small LANs and requires very little knowledge for the installation and configuration steps. For professional use on large networks, Pi-hole and eBlocker (in particular) impress with their more detailed configuration options.

In common, all solutions do their work virtually transparently and, with the exception of upribox in Ninja mode, do not affect latency or speed when you browse the Internet. The ideal solution therefore depends primarily on your requirements and preferences. ∎∎∎

and trackers. Ninja mode, which you activate in the right column, enables browsing over the Tor network [7]. However, the data throughput is then significantly lower than in silent mode.

If content is no longer displayed correctly in one of the two protection modes, click the radio button in the left column to disable all protection mechanisms for the workstation. Clicking on one of the device names tells the software to display the Internet usage statistics for that client.

To set up the system, click the *Configuration* link on the left; this step just involves a few simple boxes to check or uncheck to control access to upribox and

your WiFi network and to enable services such as SSH or an integrated VPN server (Figure 9). You can also enable a static IP address for the Raspberry Pi. The *Settings* link to the right of each option lets you configure the matching service.

The *Statistics* category in the vertical bar on the left of the main window opens a graphical display that shows statistics for blocked and filtered content, which you can sort by calendar week. A small table lists the most frequently filtered and blocked domains.

Conclusions

All of the solutions looked at in this article reliably perform their task of

Info

[1] AdGuard: *https://adguard.com/de/adguard-home/overview.html*

[2] eBlocker: *https://eblocker.org*

[3] Speed classes: *https://www.sdcard.org/developers/sd-standard-overview/speed-class/*

[4] balenaEtcher: *https://www.balena.io/etcher*

[5] Pi-hole: *https://pi-hole.net/*

[6] upribox: *https://github.com/usableprivacy/upribox*

[7] Tor network: *https://www.torproject.org/de/*

Table 1: Raspberry Pi Ad Blockers

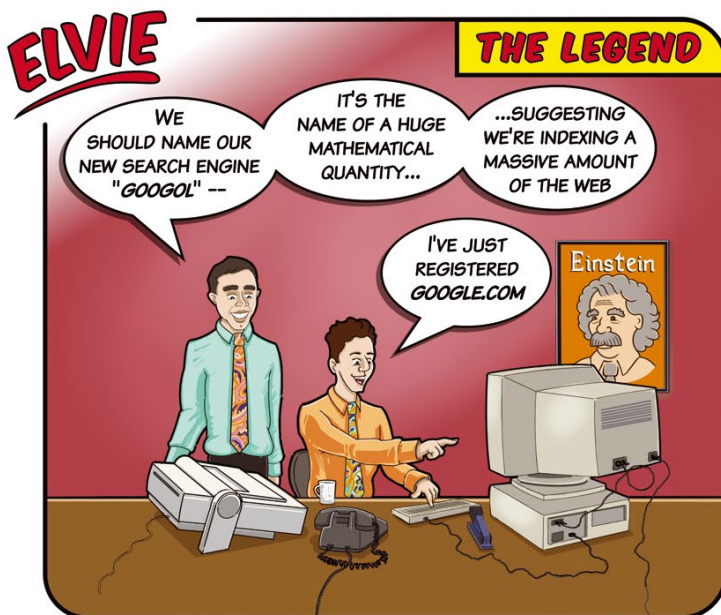| Functions	AdGuard Home	eBlocker	Pi-hole	upribox
	GPL v.3	EUPL v.1.2	EUPL v.1.2	GPL v.3
Advertisement blocking	Yes	Yes	Yes	Yes
Malware blocking	Yes	No	No	No
Web-based configuration	Yes	Yes	Yes	Yes
Automatic filter list updates	Yes	Yes	Yes	Yes
Setup wizard	Yes	Yes	Yes	Yes
Multiple configurable DNS upstream servers	Yes	No	No	No
Logging function	Yes	Yes	Yes	Yes
Graphical statistics	Yes	Yes	Yes	Yes
Additional filter list support	Yes	Yes	Yes	No
Tor network integration	No	Yes	No	Yes
VPN option integration	No	Yes	No	Yes

You push the power switch then wait. What is your computer doing as all those status messages flash by and you tap your foot, waiting for it to start? Knowing the boot process is important to knowing Linux, and an understanding of the startup steps will help you with troubleshooting. In this month's Linux Voice, we take a look at the Linux startup process – for both BIOS and UEFI systems. Also in this month's Linux Voice, modify photos at the command line with the classic ImageMagick photo tool.

Image © Olexandr Moroz, 123RF.com

LINUXVOICE▶

MADDOG'S DOGHOUSE

Linux can be a great new option for users from Windows XP owners to students with a range of interests. BY JON "MADDOG" HALL

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®.

A path to Linux

It is not often that I write about our work at the Linux Professional Institute (LPI, *lpi.org*), but there are a couple of projects spearheaded by LPI that I think are worthy of note.

The first project is Upgrade to Linux (*upgradetolinux.com*), which was originally focused on Microsoft Windows users who will be unable to upgrade to Windows 11. Either because they need more RAM or another graphics card, or they are missing the proper TPM module, these Windows users will have to upgrade their hardware or (in some cases) buy an entirely new system. This particularly impacts institutional owners such as schools, governments, and large enterprises who tend to buy "economy" systems that have smaller amounts of RAM or less expensive graphics cards – which run earlier Windows systems but will not meet the requirements to run Windows 11.

It is not only hardware that Windows 11 will abandon. There are software features of Windows 10 that will not make it to Windows 11.

Of course Microsoft points out that they will support Windows 10 for a long time, and by the time the customer really wants to upgrade their software to Windows 11 they will probably "naturally" have purchased a newer machine. But there are still millions of people using Windows XP who have not, for one reason or another, migrated to a newer version of Windows.

It is for this reason that LPI and others have advocated upgrading to Linux instead.

However, the Upgrade to Linux program does not just talk about the path from Windows 10. It talks about a lot of the advantages of moving to Linux even if you have new equipment, or (better yet) never getting stuck with Windows in the first place.

In pursuing this upgrade program, LPI realized there were a lot of myths about free software that either were never true about FOSS or might have been true at one time but are no longer. FOSS, and especially Linux, are now mainstream and are worth considering as your upgrade path.

The Upgrade to Windows program isn't limited to LPI. At this writing Ekimia, GuruTeam, Blue Lighthouse, and ownCloud have also joined, and LPI invites others.

The second initiative is Linux Clubs (*https://www.lpi.org/clubs*). Based on the great work of Stu Keroff at Minnesota's Aspen Academy, the Linux Clubs program makes it easier to start a club that makes learning about computers easier and fun. In using Linux in an environment that is not tied directly to class work, the learning takes place in an environment that encourages learning by doing.

Examples of projects can include recycling laptops and desktop systems for people (and institutions) that cannot afford new computers. Using free software on top of these systems removes the issues of license violation so often found when an older Windows system is transferred to a new user without buying a new license for it. Plus, the club participants can delve down into the software as deep as they want to go, because all of the source code is available to them.

Linux Clubs can foster interest from students who are not just interested in computers. Because Free Software covers a huge range of software tools, it allows students who have interests in photography, music, video, math, physics, and other areas of study to find, use, and improve the tools in a wide range of areas at low or no cost.

The Linux Clubs program reminds me of how I got started in computers in 1969 while a student at Drexel University. I was studying electrical engineering and having a difficult time (besides almost being electrocuted by 13,600 volts and 800 amps) when I took a correspondence course in how to program an IBM 1130 computer in Fortran as a co-op student. After returning to Drexel to continue my formal education, I found a small electrical engineering lab that had some Digital PDP-8 computers in them. By reading books and practicing, I learned how to program these machines in assembly language. Using software that was freely available for the cost of copying from the Digital Equipment Corporation Users' Society (DECUS), I gradually transferred from electrical engineering to data processing (we did not have "computer science" back in those days … it was more like "computer black magic"). Eventually I started the Drexel Computer Club, which helped to start my career both as a programmer and as an educator.

LPI has expanded the concept of Linux Clubs by offering free training materials and showing potential Linux Club leaders how to get funding for the hardware that might be desired for the club participants.

I like both of these projects, but I am particularly happy that Upgrade to Linux evolved from a rant that I made one day to the LPI staff.

Carpe Rantum. ■■■

Understanding the Linux startup process

Boot Camp

If you want to troubleshoot startup issues, you need a clear understanding of how Linux boots. BY ALI IMRAN NAGORI

This article explains the process through which the Linux kernel loads. In a more technical sense, it explains how the kernel gets into memory and what it actually does until the first user process begins its execution.

Boot Processes

At a very high level, the Linux boot processes pass through the five steps described below. Some texts can describe it in four or six steps; however, the basic mechanism remains the same. The boot process starts at the moment either when you start a shutdown system or restart an already running system.

Irrespective of the computer or OS standard ("IBM-compatible"), computer systems use either the classic BIOS/MBR [1] boot process or the modern UEFI/GPT boot process. UEFI firmware and conventional BIOS both manage system boot, but the boot method and settings differ for each.

BIOS-Based Booting

The classic BIOS/MBR boot process (as shown in Figure 1) involves five steps: the BIOS POST, loading the MBR, the bootloader phase, the kernel phase, and the startup process.

Step 1: The BIOS POST

The BIOS is located on a chip on the motherboard of a computer in read-only or flash memory and performs the Power-On Self-Test (POST) operation. This is not special to Linux itself. It is related to the hardware initialization mechanism and is similar among all operating systems. The POST is a series of tests to ensure that the computer's hardware is functioning properly. Basically, it does hardware verification in terms of availability and integrity, as well

as checking things such as the BIOS itself, CPU registers, RAM, and everything else the computer requires. During the POST operation, the system displays system messages and installs the required video drivers if needed [2].

Step 2: Loading the MBR

After the first boot-up checks have been completed, the BIOS will find the primary disk among the available disks and scan the first 512 bytes. Normally, the boot device is found in the Master Boot Record (MBR). The MBR (Figure 2) is a 512-byte area that contains 446 bytes of bootloader code, a 64-byte partition table, and the final two bytes for the boot signature. Initially, the BIOS is not aware of filesystems. As a result, the MBR is stored in a specific location on the disk, which is the first sector of the boot disk. Note that there is an MBR partition on every hard disk.

Step 3: The Bootloader Phase

After loading the MBR, the system executes the 446 bytes of the primary bootloader. This code is stored at the start of the MBR. The boot sector of the boot disk is located at this location. The bootloader code or the bootstrap code is usually called the Stage 1 bootloader. This code is too small to do any major function like calling a kernel. However, it can simplify things by locating and loading another set of blocks from the partition that contains all the code needed for loading the kernel. This code is actually the Stage 2 bootloader.

The most common Stage 2 bootloader for Linux systems is GNU GRand Unified Bootloader (GRUB). However, there are a number of other
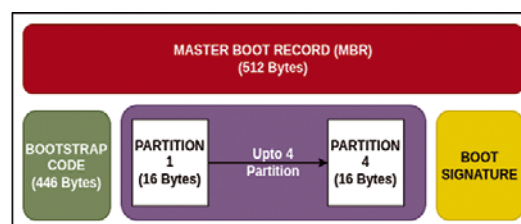

Figure 1: The traditional BIOS booting sequence.

Figure 2: The MBR Block layout.

options that are not dependent on a certain OS. The main purpose of the Stage 2 bootloader program is to locate the Linux kernel code (a disk image), which is usually inside the `/boot` filesystem [3], decompress it, read it into memory as per the GRUB configuration file, and transfer control of the system over to the kernel to move on the boot process.

Step 4: The Kernel Phase
Once the kernel starts running, it initializes the system's hardware to determine what devices are present. Also, it initializes the system's device drivers, which allow it to communicate with the hardware. Further, it sets up the system's memory management, and then it mounts the root filesystem, which contains the root directory of the system. Finally, it starts the init process from `/sbin/init`, which is the systemd process. Systemd is a replacement for the old SysVinit process and is the first userspace process to run on the system with the PID of 1.

Step 5: The Startup Process
At this point, systemd manages the user space and brings the system up to an operational state. For example, it mounts filesystems and starts other services [4].

UEFI-Based Booting
For UEFI-based systems (Figure 3), the five steps include the UEFI POST, booting with ESP, UEFI Secure Boot, the bootloader phase, and the kernel phase.

Step 1: The UEFI POST
The UEFI firmware of the system does a POST and then finds peripheral devices and the hard disk and sets them up [5].

Step 2: Booting with ESP
The UEFI utilizes a partition called EFI, which contains a boot code, to boot the system. The EFI System Partition (ESP) is a special partition on the GPT that stores EFI programs such as the bootloaders and utility software. Usually, the ESP mounts at `/boot/efi`. Therefore, `/boot/efi/EFI` is likely the beginning point for many of the EFI directory trees. The UEFI system looks for a partition with a unique GUID that distinguishes it as being the ESP. When more than one boot device (loadable kernel) is present, the UEFI boot manager chooses which ESP to utilize depending on the set boot manager order. Instead of using the predetermined order, you may choose an operating system to boot. You can specify your choice using the `efibootmgr` program.

Step 3: UEFI Secure Boot
Optionally, the UEFI boot manager makes sure to verify whether Secure Boot is turned on or off. In case Secure Boot is not set, the boot manager will cause the ESP to use the GRUB bootloader.

Other than that, the boot manager will ask the bootloader for a certificate. Then, the boot manager checks it against the keys in the UEFI Secure Boot key database before running the bootloader. Practically, to register the binary, most Linux systems utilize a small signed binary (Red Hat uses the *shim* package, for example). Additionally, the GRUB 2 loader can also be signed with a key that is user-managed via a Machine Owner Key (MOK) list. All boot binaries, including UEFI firmware drivers, EFI programs, and the OS, must have their digital signature validated by the firmware. Only after this step can the computer boot. This protects the OS from possible threats such as pre-boot malware, rootkits, and software upgrades that aren't secure [6].

Step 4: The Bootloader Phase
The GRUB 2 EFI boot loads after a successful Secure Boot verification process. The GRUB 2 bootloader, in turn, validates the kernel. In a nutshell, a bootloader is the very first piece of software that a computer's boot process activates. It is in charge of loading and handing off control to an OS kernel such as Linux.

Step 5: The Kernel Phase
The last step is for the bootloader to load the OS (the Linux kernel). The kernel then sets the stage for the rest of the OS initialization process. The post-kernel processes remain the same for UEFI as that for BIOS.

UEFI vs. BIOS
Modern PCs use UEFI firmware because it supports hard drives beyond 2TB, boots more quickly, and has better security features. UEFI is better than BIOS. However, not all motherboards support it. Many cheaper motherboards still implement BIOS configuration. Also, the initialization of UEFI firmware platforms follows a defined approach. By doing so, UEFI firmware modification becomes independent of any specific manufacturer [7].

For the UEFI-specific platform, the firmware doesn't need to worry about initializing the devices that play a more hands-on role in the boot process. Rather, these devices are initialized within the OS. As a result, this task can be done in a matter of two seconds. In the past, the same task would need as much as 10 seconds to work
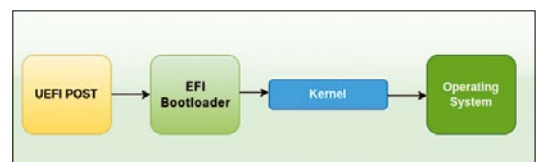

Figure 3: The UEFI booting sequence.

in the firmware. Operating systems such as Windows 8 may benefit from fewer steps when loading the OS via UEFI. Consequently, this results in a much lower boot time [8].

Conclusion

In this article, I have succinctly covered the boot process on Linux OS. I have tried to cover the two most used approaches: BIOS and UEFI. Although lengthy, this is actually a very brief introduction to the boot process. Covering every aspect and defining every term involving the boot process would require a large book. Instead, I have tried to showcase a basic idea of how the boot process flow works.

Also, the boot process is continuously evolving. Most modern systems are working on UEFI platforms with backward support for legacy BIOS. Moreover, current UEFI-compliant firmware is

quite similar to a mini operating system. In contrast, the legacy BIOS firmware has many limited functionalities. ∎∎∎

Info

[1] BIOS/MBR: *https://neosmart.net/wiki/ mbr-boot-process*

[2] Ghori, Asghar. *RHCSA Red Hat Enterprise Linux 8: Training and Exam Preparation Guide (EX200)*, First Edition. Endeavor Technologies, 2020

[3] /boot: *https://refspecs.linuxfoundation.org/ FHS_3.0/fhs/ch03s05.html*.

[4] "Startup process: *https://docs.oracle.com/ en/operating-systems/oracle-linux/8/ osmanage/osmanage-WorkingWiththe GRUB2Bootloaderand ConfiguringBootServices.html#ol-bootconf*

[5] UEFI POST: *https://wiki.archlinux.org/title/ Arch_boot_process*

[6] Comparing UEFI and BIOS: *https://blog.knoldus.com/uefi-v-s-bios*

[7] Using UEFI: *https://wiki.osdev.org/UEFI*

[8] UEFI deployment: *https://uefi.org/sites/ default/files/resources/UEFI_on_Dell%20Biz-Client_Platforms.pdf*

The Author

Ali Imran Nagori is a technical writer and Linux enthusiast who loves to write about Linux system administration and related technologies. He blogs at *tecofers.com*. You can connect with him on LinkedIn.
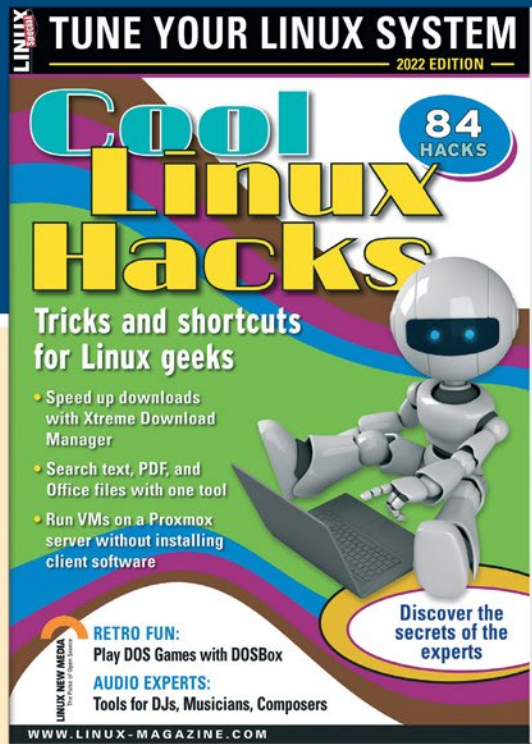
FOSSPicks

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

Graham has been playing with the source code to the amazing Mutable Instruments Eurorack modules. The company is closing, but every firmware to every product is open source. Merci, Émilie Gillet. BY GRAHAM MORRISON

Music notation

MuseScore 4

Since last looking at Muse-Score 3, the Muse project has seen a flurry of activity. Most significantly, its parent Muse Group bought the audio editor, Audacity, and created a furor over plans to add telemetry to it. They eventually relented, but the bad taste remained, requiring Muse Group to be careful with major future releases. MuseScore has suffered similar criticisms, especially with its attempts to funnel users into an online service and training portal. Because MuseScore is open source like Audacity, anyone is free to fork the project to remove the parts they don't like. It's also a project with a lot to like. There are very few professional quality music-notation applications, and none of MuseScore's caliber that are open source. Like Audacity, this makes it essential for music students, eager learners, and budget composers. Version 4 strengthens MuseScore both as a notation powerhouse and as an open source ambassador project without giving too much usability away to the money funnel.

You will first notice the music notation rendering. Previous versions were always better than another great open source alternative, LilyPond, but not by much, and it paled in comparison to its expensive proprietary alternatives. Not any more: Both the rendering and the spacing between notes has been overhauled, and musical scores are now beautifully rendered both on screen and in print. MuseScore's notation font has been improved, and the alignment of elements outside the score with the notes themselves is now pixel-perfect. While purely aesthetic improvements, they help push MuseScore into the top tier of music notation software, which is amazing when it's still open source. There are new light and dark color schemes, 400 new icons, a high-contrast mode, and a top bar that includes more configuration options, including the ability to choose what to show and hide. Selections while dragging the mouse are now intelligent and context sensitive and will only show options relevant to the complete set of selected objects. Plus you can now make gradual changes to the tempo.

In the background, a new audio engine includes support for VST3 plugins, letting you send scores directly to industry-standard virtual instruments, orchestral sample libraries, and effects, Linux compatibility permitting. SoundFonts (called MS Basic) remain supported and are the default. A new mixer helps control tracks, volumes, panorama amounts, and which tracks play which instrument. If you're prepared to leave the open source world, Muse Sounds, a new orchestral sound bank, transforms the sound output in MuseScore from pianola to professional. It's a separate 14GB download and offers much higher quality, multilayered and multi-instrument audio samples using Muse Playback Events rather than MIDI over VST. It generates naturally dynamic and fully fledged orchestral music – a brilliant way to compose music directly from the editor.

As with most aspects of MuseScore, the open source improvements and benefits massively outweigh the sometimes clunky account and plugin integration. A professional piece of notation software, you can install it and use it for free on as many devices as desired.
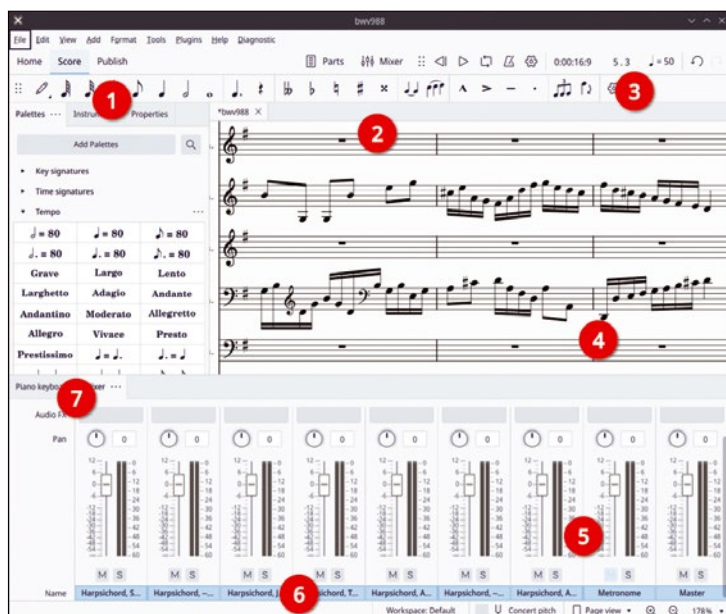

1. **New icons:** The GUI overhaul starts with beautiful icons for the toolbar … 2. **Notation:** … and continues with the beautiful rendering and spacing of the score. 3. **MusicXML:** Industry standard MusicXML files and MIDI files can be imported and rendered as a score. 4. **Annotations:** Guides outside the notes now line up with the notes. 5. **New mixer:** Besides the new mixer, a new music system (not open source) optionally can be installed for free. 6. **VST instruments:** Alongside SoundFonts, the output can be sent to VST samplers and synthesizers for a much more professional sound. 7. **Tempo:** Playback speed can now be much more finely controlled.

Project Website
https://musescore.org/en

KDE Plasma tiling

KZones

The KDE Plasma desktop has become fertile territory for people wanting to experiment with tiling window managers. This is because while KDE is a fully fledged modern desktop that will feel familiar to anyone, it's also supremely configurable. Its KWin window manager can be scripted to do almost anything, and there are several high-profile and easy-to-install scripts that will change the default standard desktop window behavior. We've looked at a couple in the past, with the best being the austerely named "Tiling Extension." With this script enabled, your application windows would automatically snap and expand to one of several layouts, filling your entire screen and even removing window borders. It used the same shortcuts and layouts as the native tiling window manager i3 and was a brilliant solution for those of us unwilling to give up all modernity in the search for tiling efficiency. Unfortunately, the Tiling Extension script is no longer being developed or supported, and the project has been archived. This has led to a glut of alternatives trying to win its user base, including Exquisite, which we looked at previously, and KZones, a beautiful solution that includes graphical configuration, on-screen prompts, editable zones, and keyboard shortcuts. All of this is accessible from its System Settings panel, which lets you change how zones are triggered and how you drag windows into those zones to activate them. The layouts are easy to create yourself and include percentages and positions for each column or row you want to create. Shortcuts can be assigned via KDE's global settings. It's the easiest on-ramp to tiling
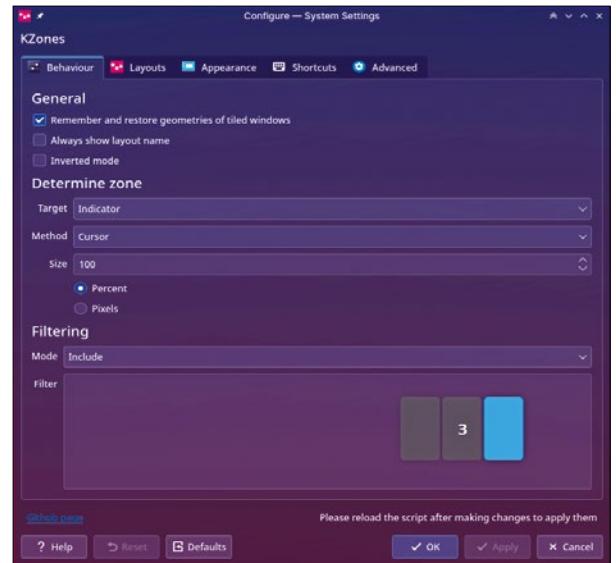
The main difference between KZones and a real tiling window manager is that there's no concept of a master window around which every other panel arranges itself.

we've seen and a brilliant option for KDE Plasma users who want to make the most of their desktop space without committing entirely to tiling.

Project Website
https://github.com/gerritdevriese/kzones

Office manager

Kraft

Creating and maintaining the documentation associated with running a business isn't normally something you can get too geeky about. It usually involves the sober writing of invoices, customer contact details, quotes, and calculations, which is probably why so many commercial products take advantage of this monotony with their proprietary solutions. Kraft lets you accomplish all of this while letting you explore your geeky curiosity and support for open source. Kraft 1.0 is a Qt application that integrates perfectly with KDE Plasma, but it looks just as good from other desktops, too. Its 1.0 tag is also something of a misnomer. Despite only being released at the end of 2022, Kraft has been in development since 2004, with lots of public 0.1 releases along the way. This means it arrives fully formed and stable enough for professional use on whichever desktop you choose. Kraft is a content management system for business documents. It lets you create them, update them, and keep track of them across a timeline. The documents are created within the app, with their type selected from a drop-down list that can include your own types. Select an invoice, for example, and you can use KDE's built-in KAddressBook to assign a customer and add notes for how the document will appear on the virtual whiteboard. Documents are built from editable templates, and the invoice, for example, lets you add and edit chargeable items, with hourly rates and tax calculations, in a way that can be given to customers and kept and processed as part of your own records. The whiteboard is an easy-to-view list of outstanding documents that makes those that are current easy
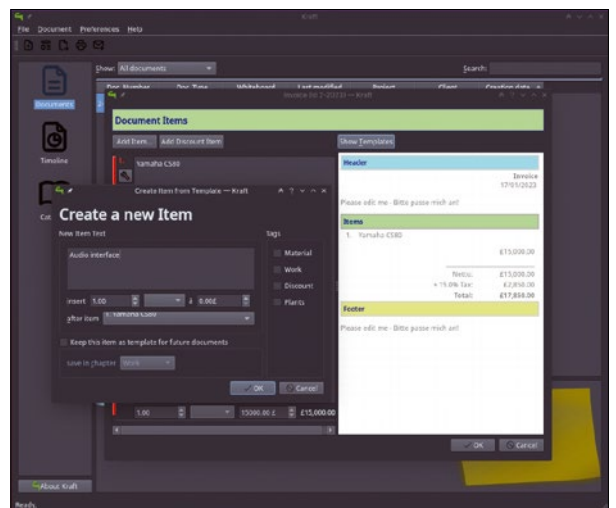
Kraft can create and manage invoices, offers, delivery receipts, and many other types of documents, managing them across time and versions with a database back end.

to access. Older documents can be opened from the timeline or from the *Documents* view, which lists every document in the database and lets you filter by tags, type, and text. If you run a small business, Kraft should offer everything you need.

Project Website
https://volle-kraft-voraus.de/Main/Details

Terminal copy and paste

Clipboard

O n the desktop, the clipboard is entirely taken for granted. Many of us now use Ctrl+C in one application and Ctrl+V in another to copy text from one place to the next without even thinking about it. Those commands, along with Ctrl+X to cut, have been consigned to muscle memory. But it's not quite so simple on the command line. Historically, the command line didn't need this kind of copy and paste functionality because it was already purpose built for manipulating streams of data. Real hackers were supposed to pipe and filter output between commands rather than crudely copy and paste without automation. Ever since desktops arrived, and terminals moved from teletypes to windows, the two have coexisted

with very little option for interaction, a situation made even worse when you're connected remotely. The mouse's handy drag-and-select, along with middle-mouse-button paste, has helped bridge the gap, as have tools such as `xclip` for copying output to the desktop clipboard, but there's still been very little modern innovation, and that's something Clipboard is hoping to provide. At the heart of its functionality is the `clipboard` command, which can also be abbreviated to `cp`. It's remarkably straightforward to use. You simply run the command directly or wrap other commands around it. Type `cb copy README`, for example, and the contents of the README file will be copied to an internal clipboard. Adding a number to the end of
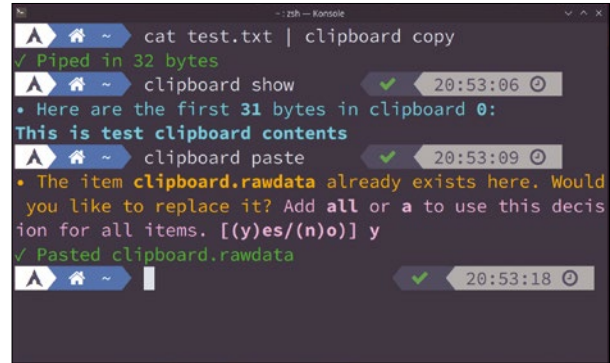
Clipboard can copy and paste across two systems using the SSHFS utility and a shared directory location.

the command will use that numbered clipboard instead. Typing `cb paste` will output the clipboard contents to the same or a different file, which you can select from a prompt, and the `show` command will display the contents of all your clipboards. Environment variables can be set to remove all prompting and also to create a persistent directory for clipboards, which is useful if you want to share them with a remote session. Best of all, it interacts directly with the X11 clipboard with no further configuration, and Wayland support is coming.

Project Website
https://github.com/Slackadays/Clipboard

Command runner

just

C ommand runners are curious things because there are so many ways to automate the running of commands. Shell scripts are the simplest and probably the most widely used. If you can type commands into a shell, you can type them into a file and run this separately. But they're also prone to errors, and not particularly resilient to changing environments, location, or systems. At the other end of the scale is a makefile, which has a heavy syntax of its own for running and controlling the commands it runs. The strangely named `just` is just another example, and it bridges the best of both worlds. It makes command automation as easy as writing a script, but also as potentially programmable as a makefile.

The commands themselves are grouped together into a YAML-like text file called a `justfile`. These recipes are easy to build and read but are also deceptively powerful. At their simplest, you create a label followed by an executable command, and running the `just` command followed by the label will run that command. All output is included, as are comments and other feedback from the recipe. But this is just the beginning. Recipes can also include settings, arguments, variables, substitution, and even built-in functions. You can also include conditional expressions, and these are useful for iteration and parsing output and usually more generic than their shell script alternatives. All of this is beautifully documented in the project's main README file.
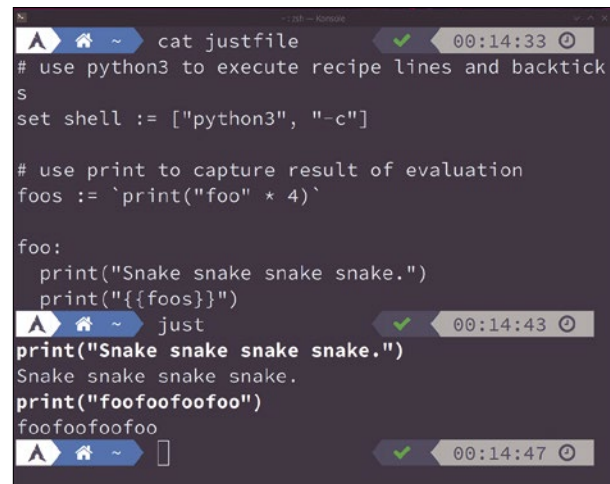
Create batch control scripts, conditional scripts, testing scripts, and even run and parse native Python commands with `just`.

Despite the breadth of functions on offer, it's easy to understand and use yourself. Unlike a makefile, which requires encyclopedic knowledge of the syntax before you even start, `just` is fun and easy to use. If you ever find yourself running the same old commands over and over, perhaps on different machines on different distributions, `just` can be a more practical and pleasant experience.

Project Website
https://github.com/casey/just

CD ripper
Whipper

The name of this command line application seems reminiscent of Devo's definitive 1980s song, "Whip It." This isn't perhaps surprising considering we're dealing with a distinctly 1980s piece of high technology, the audio compact disc. But it's more likely, and less interesting, that the name comes from the act of ripping – copying the data from an audio CD to your computer. Your first thought might be why are CD rippers still relevant in the third decade of the 21st century, and there are two answers to this. The first is that despite a plethora of streaming services, very few serve digital audio at CD quality. Most offer streaming only via lossy compression, and CD audio delivers a gloriously uncompressed 16-bit value, 44,100 times a second. The second is that you can't always stream or buy everything that was available on a CD. Even if you can, it's probably been re-mastered beyond recognition. This makes many of the original CD recordings the definitive article and still worth ripping.

The ripping process itself is also surprisingly complicated. This is because while the audio data is digitally encoded on the disc, the optical diodes used to read the data are analog and can introduce read errors. While there is some error correction, it's not always enough to guarantee the integrity of the audio data, which means multiple readings and averaging are often required. And this is what

If you can find a CD drive, Whipper can rip your favorite CD recording.

Whipper does, "aiming for accuracy over speed." It can detect hidden tracks, read offsets, perform re-encoding, and correctly name and tag the extracted music files. It can also use AccurateRip's drive offset database to set drive offset values to correct known drive misalignment, although you'll need to be using one of the drives already listed. And that's likely to be the biggest problem if you want to finally rip your audio collection – finding a modern computer with an optical drive that can read audio discs.

Project Website
https://github.com/whipper-team/whipper

Terminal charts
YouPlot

Considering how much data we process on the command line, it's surprising how few tools there are to help with data visualization. Being composed almost entirely of ASCII characters, the command line has its own limitations when it comes to output, and that can be a challenge. But we can surely do better than throwing a few asterisks across the display. And with YouPlot, you can. YouPlot is a simple command-line tool that's easily installed with Ruby's RubyGems package manager and just as easy to use. First, despite installing a project called "YouPlot," the executable is `uplot`, presumably to lessen the typing burden while making a pin. The command can take its input from either the standard input or from a file. While the input format is based on tab-separated values (TSV), it's both flexible and configurable. By default, it will make the table with the first column representing the x-axis and the second column the y-axis. The space character is the default delimiter, but this can be changed to tab or any other character, as can which columns are used for which axis. A single column can also be parsed and used in either the line or bar graphs.

YouPlot currently supports eight different charting methods, including bar plot, histogram, line plot, scatter chart, and a box plot. They all look rather wonderful, and there's something magical about using `curl` to grab an online table live, piping its output into `cut` to strip extraneous data, and pushing that into `uplot`. The output looks fantastic and is genuinely useful for quick and accurate data perusal from the command line. The README file that
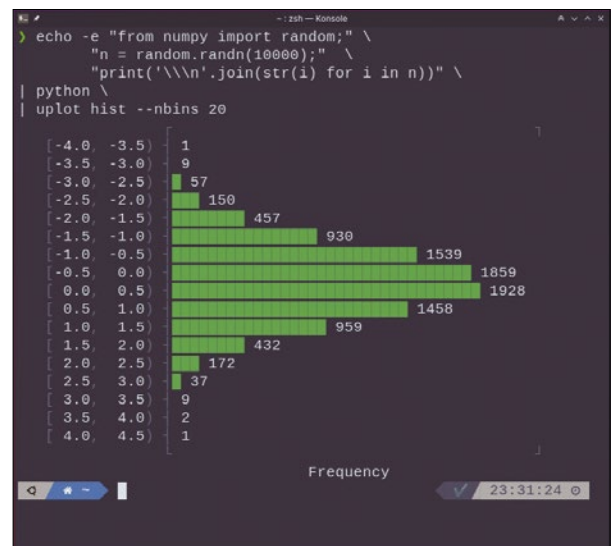
Win friends and influence people with your mighty command-line charting abilities

accompanies the project is equally effective, showing plenty of illustrated examples and even includes charts that are made from live Python output. It's a brilliant little tool, and one that can change your mind about how capable and informative command-line output can be.

Project Website
https://github.com/red-data-tools/YouPlot

Neural synthesis and effects

DDSP-VST

DSP is an utterly unique audio synthesizer and effect plugin that will work with your favorite audio editor or workstation, including Audacity, Ardour, and Bitwig. It's unique in a couple of ways, with the first being that there's no official Linux compatibility. It can be built on Linux, however, by applying a simple patch currently under review in the project's GitHub repository. This will generate a Linux-native VST library that can be dropped into your VST path (usually `/usr/lib/vst` or `$HOME/vst`). With that done, you're free to explore the other aspect that makes DDSP unique, and that's a sound model derived from a neural network. Differentiable Digital Signal Processing is what the DDSP name represents, an academic research project and upstream computational library that generates sounds from models. This model learns from example recordings to create an instrument that preconfigures a collection of linear filters and sinusoidal oscillators to generate a sound closely related to the example recordings. DDSP is differentiable because it's able to adapt to different datasets, and thus, lots of different sound characteristics, from natural acoustic sounds to purely fictional frequency curves. I don't understand any of this either, but it's easy to hear that the output and control DDSP offers is utterly unique.

Eleven presets are included for the bundled pre-trained models, including flute, trumpet, violin, and a collection of vowels. The synth accepts MIDI input and generates a tone, but this can be manipulated like a physical instrument by dragging a reticule across the grid of pitches. This dynamically adjusts the complex series of filters generating the sound to emulate different mouth shapes and positions, for
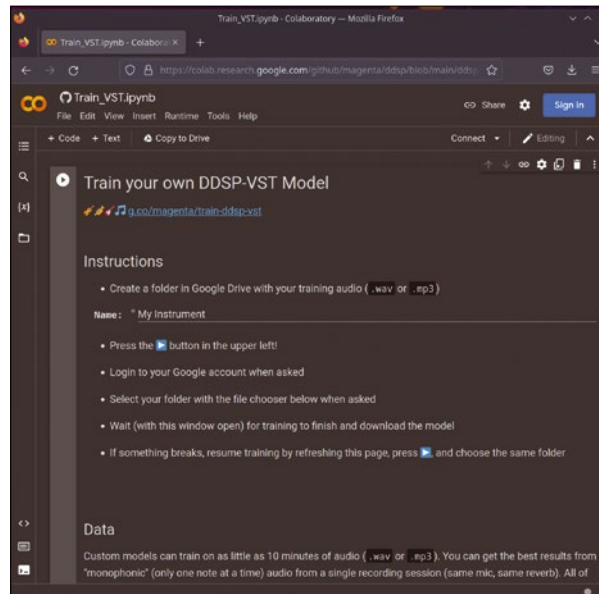
The project includes its own web portal where you can drop your own sound sources to general neural network models that can be loaded into DDSP-VST.

example, or different playing positions for strings. When this reticule is controlled externally, such as with a touch surface or a MIDI wind instrument, the output can sound incredibly organic and natural, even if it doesn't always sound too similar to the source recordings. This is due to the complicated interrelationship between all the elements built into the model being controlled and calculated in real time, and not at all like what you get from a more traditional subtractive synthesizer. It's also monophonic, which means you'll need to combine more than one instance to play more than one note at a time, but that's completely normal for a lead instrument or bass sound.

The other interesting aspect to this plugin is that it can be used as an audio effect to process any audio sent through it. Cleverly, the pitch and amplitude of the incoming audio are automatically detected and used to drive the model characteristics applied to the incoming sound. The sound can then be manipulated in exactly the same way as the synthesizer, using the rectangular reticule to change the model parameters and adjusting the amount of harmonics, noise, and pitch shift. The noise alone can create some great drum sounds, while other inputs can generate output that sounds anywhere between digital noise and scary whispering. Finally, a layer of professional sheen can be added to the output with inbuilt reverb. It's definitely experimental, but also unique and a lot of fun to play with.

Project Website
https://github.com/magenta/ddsp-vst/blob/main/README.md
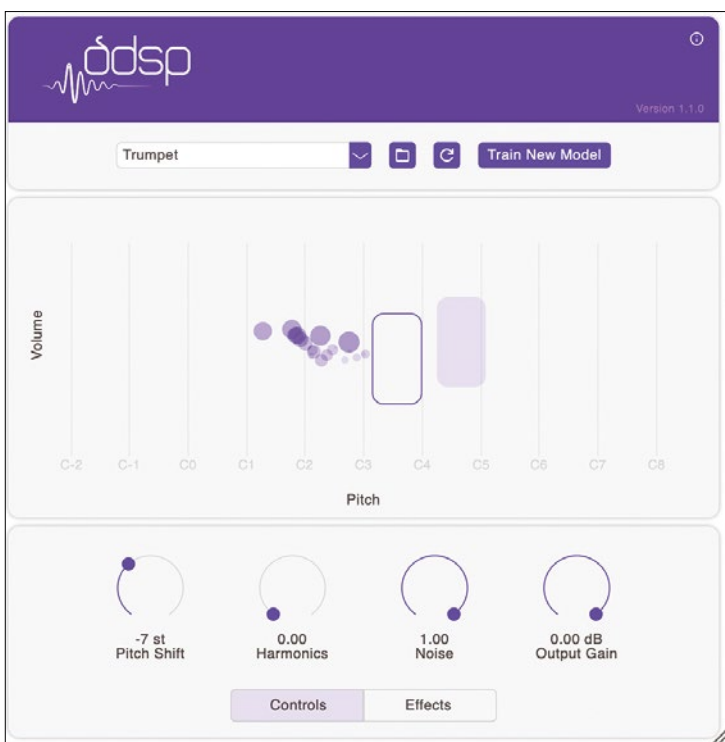
DDSP generates uniquely organic and evolving sounds that can be controlled like acoustic instruments.
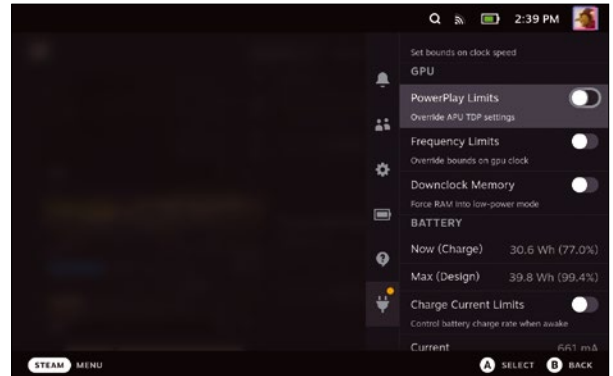
Decky Loader

Valve's Steam Deck has undoubtedly been a success, with well over one million units reportedly sold. But it's also been an interesting experiment in just what happens when the gaming public comes into contact with Linux (Arch Linux, no less!). The results have been remarkable. Perhaps it's because many gamers are used to tinkering with their Windows system to get every drop of extra performance from their hardware and installing hundreds of mods, or perhaps it's because they're realizing the potential in a completely open system for the first time. But either way, there's now a flourishing ecosystem of third-party plugins, skins, intro animations, and other add-ons that augment the factory Steam

Deck functionality. The only problem is that, just like the mod scene itself, all those dozens of projects and their components soon become difficult to manage, configure, and keep up-to-date. Decky Loader solves this problem.

Decky Loader is a plugin that sits within Valve's default right-side Steam Deck UI menu and provides quick access to menu themes, system sounds, screen saturation, PowerTools system settings, and a plugin system that can add many more options. This menu is always available, which means settings can be accessed while you're in a game. It still feels unusual seeing a general project have install instructions that include switching to the Plasma desktop and running a file from

Decky Loader does need to be installed from desktop mode, but it's then fully self-contained and a brilliant add-on for your Steam Deck.

Dolphin, but it works without a hitch. When the Steam Deck UI is started, Decky is then the bottom option in the right Steam Deck UI menu. Any plugins that are already installed will be listed and their settings values accessible, but you can also install plugins directly from the integrated "store" link. From here you can directly download plugins, including CSS themes, a notebook, MPRIS music control, the PowerTools add-on, and badges for your games to show ProtonDB compatibility. It feels fully integrated with the Steam Deck UI and is constantly being updated.

Project Website
https://github.com/SteamDeckHomebrew/decky-loader

FreedroidRPG 1.0

FreedroidRPG is an isometric role-playing game that has been around for a long time. The original "classic" version was first released in 2002 when there were very few games for Linux other than Tux Racer and BZFlag, and the classic version attained a 1.0 version number in 2003. Just before this, the project forked into a version known as the "modern" version, which included an increasing number of modifications. This modern version has been under development ever since and has just celebrated its own 1.0 version release, 20 years after the classic version did the same. To celebrate, the game is now available from Steam with a single-click installation, as well

as the AppImage you've been able to download directly from the project's web page.

Despite its age, FreedroidRPG is worth playing and is a particularly good fit for the Steam Deck's performance, casual game-play style, and portable form-factor. It has also become unintentionally retro-cool and has a story similar to Mario RPG, combined with the combat mode similar from earlier Diablo games, albeit on a sci-fi future spaceship with a penguin avatar. You control Tux with the mouse, and there's a tutorial mode to ease you into the game's mechanics. There are ranged and melee weapons, and you can take over enemy robots just like you could in Andrew Braybrook's 1980s classic,

FreedroidRPG was inspired by the Commodore 64 classic, Paradroid, and even includes a very similar font for the dialogue.

Paradroid. But there's a deeper story too, involving a destroyed world and the battle between robots and humans, with the whole game lasting around 10 hours. If that's not enough, the project welcomes all contributions, and it would make a great first project for someone who wants to get into indie game development. There's even a built-in level editor where you can create your own levels to share with the community.

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

Professional graphics and photo editing
at the command line

Image Wizardry

The free ImageMagick graphics toolbox brings the feature set of a full-blown image processor to the command line. BY CLAUDIUS GRIEGER

For more than 30 years, a group of about 30 people has been developing the ImageMagick [1] graphics toolbox and offering it for free under the Apache 2 license. It is not a single program, but a set of compact tools for the command line. In addition to a Linux version, there are variants for Windows, macOS, iOS, Android, and other operating systems. ImageMagick supports over 200 file formats and reliably processes images with resolutions in the gigapixel and terapixel range.

You can install the program collection using your choice of distribution's package manager. At the time of going to press, only Arch Linux had the latest 7.1 version on board. Debian 11 (including

Testing and Unstable) and Fedora 37 still offered version 6.9. On Debian-based systems such as Ubuntu and Linux Mint, you need to use the command from line 2 of Listing 1 to install. On Fedora, use the command from line 4. On Arch-based systems such as Manjaro, set up ImageMagick using the command from line 6.

After doing so, you will find the command-line programs listed in Table 1 on your computer. Many instructions on the Internet still refer to old commands where users always had to type `magick` followed by the command string. The newer versions have dropped this, meaning that `magick convert` has been simplified to a plain `convert`.

Everyday Commands
You can use the `identify IMAGE FILE` command to initially identify an image without opening it outside the terminal with your distribution's file manager, or you can use the `display` command. The output then looks like line 2 of Listing 2, for example. If you want to limit the output to the height, width, and file names, add options to the command as shown in line 3; `\n` stands for a line break here. This is how you

Listing 1: Installation

```
01 ### Debian and derivatives
02 $ sudo apt install imagemagick
03 ### Fedora
04 $ sudo dnf install imagemagick
05 ### Arch Linux and derivatives
06 $ sudo pacman -S imagemagick
```

Table 1: Tool Overview

Command	Function
`convert`	Main tool for converting, scaling, blurring, sharpening, cropping, rotating, and more
`mogrify`	Like `convert`, but replaces the source file
`identify`	Reads out various image parameters including the Exif data
`composite`	Overlaps images
`assembly`	Combines multiple images into one target file
`compare`	Marks differences between images in the destination file or outputs them as values
`stream`	Copies pixel data from one image to another
`display`	Displays images via an X server as a pop-up
`import`	Creates screenshots in the X server
`conjure`	Runs scripts in the Magick Scripting Language

Listing 2: Retrieve Image Data

```
01 $  identify image.jpg

02 image.jpg JPEG 811x664 811x664+0+0 8-bit sRGB 62201B 0.000u 0:00.000

03 $ identify -format "%f width: %w Height: %h Format: %m \n" image.jpg

04 $ identify *.jpg >> TARGET.txt
```

Listing 3: Mass Rename

```
$ for i in *.jpg; do \
  convert "$i" "${i%.jpg}.png"; \
done
```

can create photo lists for admin purposes, for example. The command from line 4 appends the image data to the `TARGET.txt` file.

Converting and More

You can use the versatile `convert` command for converting images to another file format, renaming, cropping, and resizing, as in:

```
convert SOURCE TARGET
```

This always creates a new target file, leaving the original image untouched. The counterpart `mogrify` uses almost identical syntax, but processes the source file directly.

If needed, you can convert from one file format to another and rename the file at the same time. To change only the format, use:

```
convert image.jpg image.png
```

If you only want to change the file name, type:

```
convert image.jpg photo.jpg
```

In combination the command would be, say:

```
convert image.jpg photo.png
```

If needed, you can also convert all files of a certain type in a single action. There are several possible approaches to choose from. The command

```
convert *.jpg *.png
```

converts all JPEG files to the PNG target format, but without using the names. The target files are then named `*-1.png`, `*-2.png`, and so on. However, you can easily change this with terminal commands (Listing 3).

If you want to rotate or resize images, `mogrify` often proves useful because it converts the source file directly. For example, if all the images in a folder are portrait but should be displayed in landscape format, you can rotate them all clockwise in one fell swoop using the command

```
mogrify -rotate "90" *.png
```

or the other way around using

```
mogrify -rotate "-90" *.png
```

Parameters such as `-geometry 800x600` or `-resize 50%` are used to reduce the size of the images.

You can create thumbnails of all the JPEGs in a folder using a batch command such as the one in line 1 of Listing 4. It names the target files `preview-1.jpg`, `preview-2.jpg`, and so on. If you want to take the file name and save it in a separate folder (e.g., `thumbs/`), the command might look like the one in line 2. However, to do this, you first need to create the `thumbs/` folder. If you only want to edit the dimensions of an image page, enter, say, `200x` or `x200` after `--resize` instead of `200x200`. If you use the `-thumbnail` parameter instead of `-resize`, `convert` automatically deletes comments and color profiles from the metadata (line 3).

Listing 4: Create Thumbnails

```
01 $ convert *.jpg -resize 200x200 preview.jpg

02 $ for i in *.jpg ; do convert "$i" -resize 200x200 "thumbs/$i"; done

03 $ convert *.jpg -thumbnail 200x200 preview.jpg

04 $ convert -delay 100 *.jpg animation.gif
```


Figure 1: To apply a blur, use `-blur RADIUS`.
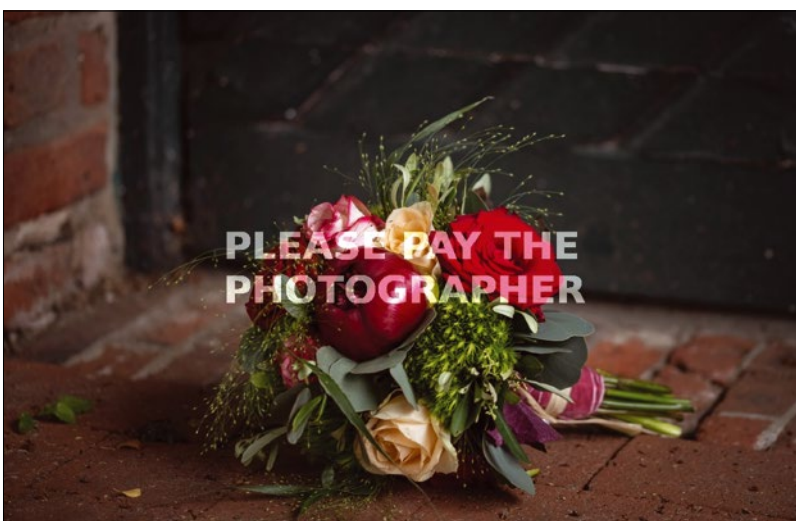
You can convert image sections into animations in a similar way. To create an animated GIF from a set of JPEG files, use the command from line 4 of Listing 4. Enter the value after `-delay` in hundredths of a second. It sets the time between image changes. If you specify `100`, as in the example, a new image appears every second.

Effects and Compression

You can do more with `convert` and `mogrify`, such as using them to integrate image effects directly. You can use `-blur RADIUS` to apply a blur with the specified radius (Figure 1). If you want your image to look more artistic, `-charcoal FACTOR`, gives you a charcoal effect (Figure 2). To simulate a colored pencil drawing instead, use `-sketch RADIUS` (Figure 3).

If you want to compress your images to save storage space, there are 25 algorithms to choose from as `-compress TYPE` options. They include bzip2, FAX, JPEG 2000, Lossless, WebP, and many others.

Use `-contrast` or `+contrast` to increase or decrease the contrast of the target image. The `-sharpen RADIUS` command increases the sharpness, with `0x1` being a good starting value. You can change the value on both sides of the `x` until you achieve the desired results. To surround your image with a frame, use

```
-border 1x1 -bordercolor 0xFF0000
```

to create a red frame with a thickness of one pixel.

Copyright and Co.

A little more complexity comes into play once you create custom shapes via ImageMagick's tools, mark images with a date or image credit, or add a watermark for online preview.

The command in Listing 5 adds a copyright label bottom left in the source image (`SouthWest` – note the capitalization of the cardinal directions). The font used here is 150-pixel-sized DejaVu Sans in white. The *xy* values that follow `-annotate` can be used to move the font pixel by pixel if necessary (e.g., `+5+20`). In the example, I left it at the original position (`0`). The correct

Figure 2: The charcoal pencil effect combines well with a soft focus.

Figure 3: This sketch effect needs further refinement.

Figure 4: Watermarks help you protect your images online.

Listing 5: Label Images

```
$ convert image.jpg -gravity SouthWest -pointsize 150 -fill white \
  -font DejaVu-Sans -annotate 0 '(c) PHOTO FROM XYZ / 2022' target.jpg
```

Listing 6: Watermark

```
$ composite -watermark 50 \
  -gravity Center \
  WATERMARK.png \
  SOURCE.jpg TARGET.jpg
```

Listing 7: Montage

```
$ montage -label %f \
  -geometry 800x600+10+10 \
  *.png imageindex.jpg
```

spelling of the installed fonts is provided by the command:

```
convert –list font
```

To set a watermark, it's best to use ImageMagick's `composite` program to blend the source image with an existing image that has an alpha channel (i.e., a transparent background) (Figure 4). The value following `–watermark` can be between `0` (transparent) and `100` (fully opaque). The `50` in the example from Listing 6 means 50 percent opacity. The command, like all others with an asterisk instead of the filename, can be applied to all files of a certain type.

Merging Images

You can use the `assemble` command to merge images into a single file. Some photographers use such images like this to capture folder contents at first glance. This can be very useful especially in the cloud or if bandwidth makes it difficult to preview images individually. The command from Listing 7 saves all PNG files of the current folder with a size of 800x600 pixels side by side in a JPEG and with defined spacing. In each case, the file name is inserted below the image (Figure 5).

Conclusions

The features mentioned in this article only scratch the surface of ImageMagick's

capabilities. The program has far more to offer, and detailed knowledge of it fills books. If you want to learn more about the extensive tool, the project homepage and the man pages of the individual tools are good places to start. The ImageMagick website also has a practical guide about drawing with the tool [2].

When working with the graphics toolbox, always bear in mind that learning a new tool will take some time. Even if you never quite master it or use it to create elaborate image compositions, ImageMagick is still a handy all-around tool in everyday life because of its batch options for converting formats, easy resizing, or embedding watermarks. ▪▪▪

Info

[1] ImageMagick: *https://imagemagick.org*

[2] Drawing with ImageMagick:

https://imagemagick.org/Usage/draw/

The Author

Claudius Grieger discovered ImageMagick at the command line more or less by accident and has been using it ever since on his Debian system for batch conversion of graphics and photos.
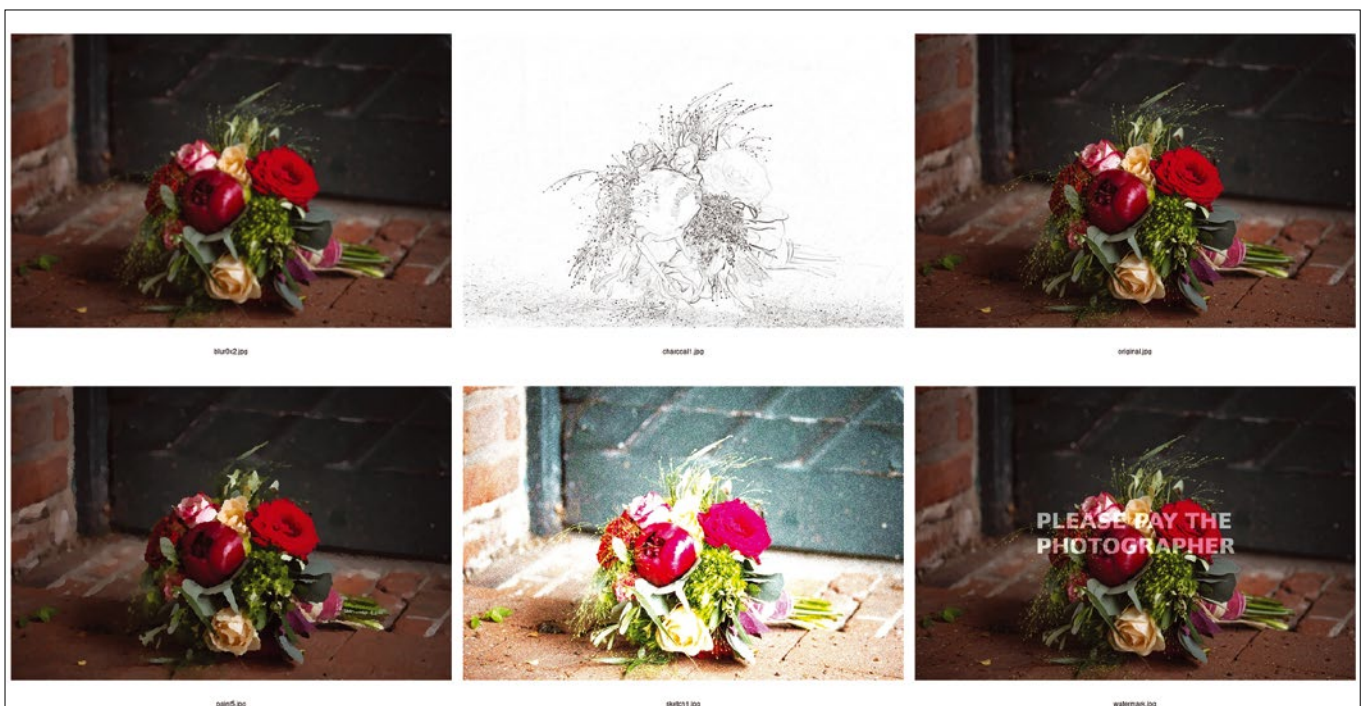
Figure 5: A labeled overview in an image helps you to save data volume in the cloud.
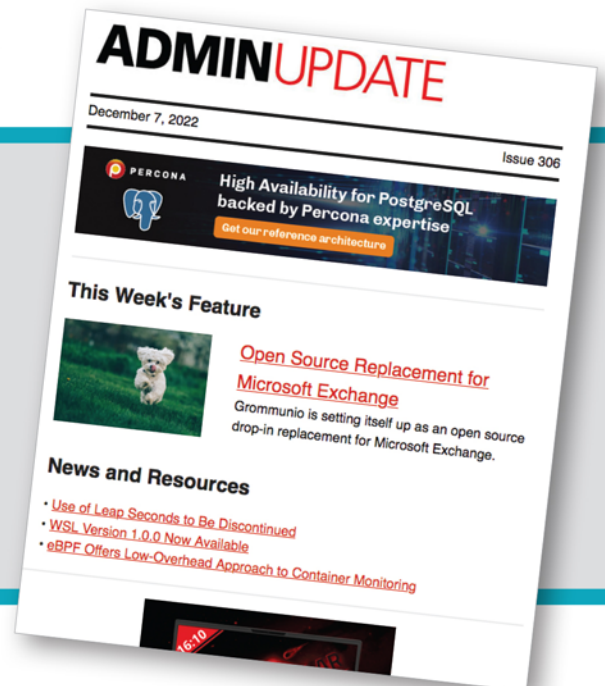
LINUX NEWSSTAND

Order online:
https://bit.ly/Linux-Newsstand

Linux Magazine is your guide to the world of Linux. Monthly issues are packed with advanced technical articles and tutorials you won't find anywhere else. Explore our full catalog of back issues for specific topics or to complete your collection.
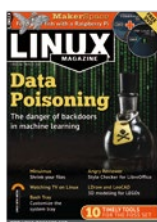
#268/March 2023

Data Poisoning

Think computers don't make mistakes? If you slip some doctored-up training samples into the mix, you can get a fancy machine-learning system to think a dog is a cat or a 1 is an 8 – and you can trigger this bad behavior through a hidden signal no one else will notice.

On the DVD: MX Linux 21.3 and Puppy Linux FossaPup 9.5

#267/February 2023

Backup

In theory, everyone could use the same backup tool, but the best way to build regular and systematic backups into your life is to find a solution that fits with your own habits and methods. This month, we preview some popular backup apps in the Linux space so you can find one that works for you.

On the DVD: Linux Mint 21 Cinnamon and Kali Linux 2022.4

#266/January 2023

Generative Adversarial Networks

What is the secret behind the recent explosion of computer art and fake videos?
One neural network lies to another neural network...

On the DVD: Ubuntu 22.10 and AlmaLinux 9.0

#265/December 2022

Quantum Computing

Most Linux users know that this futuristic technology leverages the weird power of quantum mechanics. But how does it really work? What can I do with it? Are there tools available today that will help me experiment? This month we take a deep dive into quantum computing.

On the DVD: Manjaro 21.3.7-220816 and Arch Linux 2022.10.01
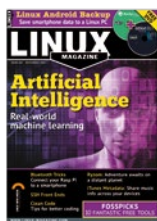
#264/November 2022

Artificial Intelligence

Machine learning remains shrouded in mystery even though it is now an integral part of our everyday world. This month we look behind the curtain at some popular techniques for supervised and unsupervised learning.

On the DVD: Debian 11.5 and Rocky Linux 9.0

#263/October 2022

Build an IoT Linux

The most amazing thing about Linux is its flexibility. Linux systems run on the biggest computers in the world – and on many of the diminutive devices that populate your home environment. If you've always wondered how developers adapt Linux to run on tiny tech, you'll appreciate this month's stories on Buildroot and the Yocto project.

On the DVD: Linux Magazine Archive issues 1-262

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 *https://www.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 *info@linux-magazine.com*.

PyCon US 2023

Date: April 19-27, 2023

Location: Salt Lake City, Utah

Website: *https://us.pycon.org/2023/*

We are so excited to welcome our community back to the Salt Palace Convention Center for the 20th Anniversary of PyCon US! The program is filled with 90+ of our community's best talks, brilliant keynote speakers, a lively Expo Hall, and our famed lightning talks on each main conference day. Following the conference days, there are 4 days of sprints that are free and open to the public.

Linux App Summit

Date: April 21-23, 2023

Location: Brno, Czech Republic and Online

Website: *https://linuxappsummit.org/*

The Linux App Summit (LAS) brings the global Linux community together to learn, collaborate, and help grow the Linux application ecosystem. Through talks, panels, and Q&A sessions, we encourage attendees to share ideas, make connections, and join our goal of building a common app ecosystem. Join us in-person or online.

DORS/CLUC 2023

Date: May 11-12, 2023

Location: Zagreb, Croatia

Website: *https://www.dorscluc.org/*

DORS/CLUC is the oldest and biggest regional conference that covers free and open source software, open standards, and Linux. During two days of talks, workshops, and fun, this event brings together students, FOSS enthusiasts, tinkerers, developers, companies, and more to learn, network, do business, and plan projects together – all with a focus on free software and Linux.

Events

KubeCon + CloudNativeCon Europe 2023	Apr. 17-21	Amsterdam, Netherlands	https://events.linuxfoundation.org/
PyCon US 2023	Apr. 19-27	Salt Lake City, Utah	https://us.pycon.org/2023/
Hybrid Cloud Conference	Apr. 20	Virtual Event	https://www.techforge.pub/events/hybrid-cloud-congress-2/
Linux App Summit 2023	Apr. 21-23	Brno, Czech Republic	https://linuxappsummit.org/
Cloud Expo Europe	May 10-11	FrankfurtFrankfurt, Germany	https://www.cloudexpoeurope.de/
DORS/CLUC 2023	May 11-12	Zagreb, Croatia	https://www.dorscluc.org/
Icinga Camp Berlin	May 17	Berlin, Germany	https://icinga.com/community/events/icinga-camp-berlin-2023/
ISC High Performance	May 21-25	Hamburg, Germany	https://www.isc-hpc.com/about-overview.html
openSUSE Conf. 2023	May 26-28	Nürnberg, Germany	https://events.opensuse.org/conferences/oSC23
CloudFest USA	May 31-Jun. 3	Austin, Texas	https://www.cloudfest.com/usa/
DrupalCon Pittsburgh	June 5-8	Pittsburgh, Pennsylvania	https://events.drupal.org/pittsburgh2023
Opensouthcode 2023	June 6-10	Málaga, Spain	https://www.opensouthcode.org/conferences/opensouthcode2023
OW2con'23	June 14-15	Paris-Chatillon, France	https://www.ow2con.org/view/2023/
Akademy	July 15-21	Thessaloniki, Greece & Online	https://akademy.kde.org/
GUADEC	July 26-31	Riga, Latvia	https://events.gnome.org/event/101/
WeAreDevelopers World Congress	July 27-28	Berlin, Germany	https://www.wearedevelopers.com/world-congress/
stackconf 2023	Sep 13-14	Berlin, Germany	https://stackconf.eu/
DrupalCon Lille 2023	Oct. 17-20	Lille, France	https://events.drupal.org/lille2023
Open Source Monitoring Conference (OSMC)	Nov 9-7	Nuremberg, Germany	https://osmc.de/

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

Issue 270 / May 2023

Green Coding

The future looks brighter if everyone pays attention to energy. Alternative sources are great, but another way to conserve greenhouse gasses and build a sustainable future is to use less. Next month we explore some techniques for more energy-efficient programming.

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: *https://bit.ly/Linux-Update*

Image © Ledda Maria Rita, 123RF.com