



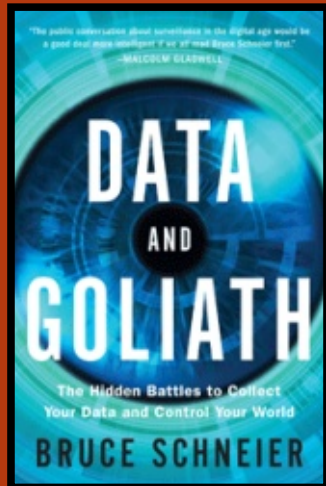
# Full Circle

THE INDEPENDENT MAGAZINE FOR THE UBUNTU LINUX COMMUNITY

ISSUE #102 - October 2015



## BOOK REVIEW



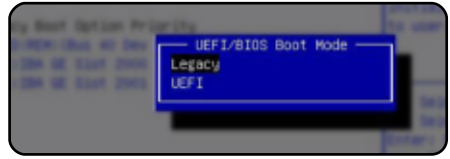
# MULTIBOOT WITH UEFI KEEP WINDOWS 10 AND USE LINUX



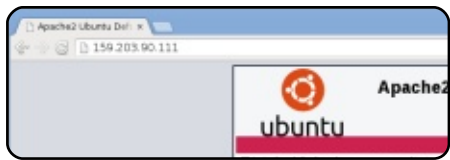
## HowTo



Python in REAL World p.13



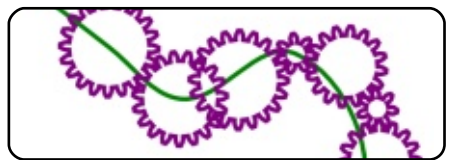
Multiboot With UEFI p.14



Website & Infrastructure p.20



Minimal Install p.25



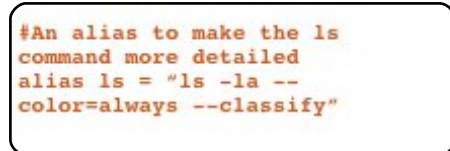
Inkscape p.34

## Graphics



# Full Circle

THE INDEPENDENT MAGAZINE FOR THE UBUNTU LINUX COMMUNITY



Command & Conquer p.10



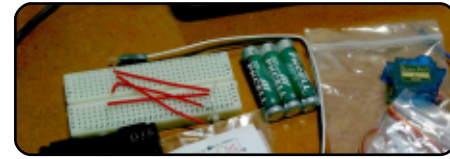
Linux Labs p.45



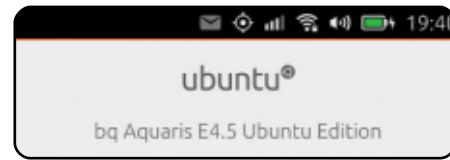
My Story p.53



Q&A p.57



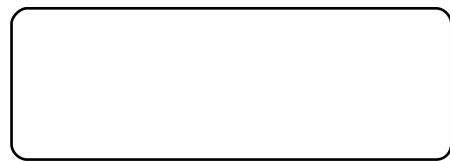
Arduino p.39



Ubuntu Phones p.49



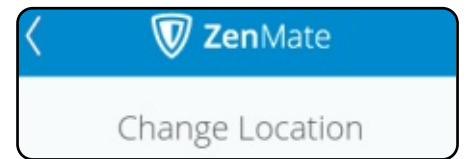
Letters p.54



Security p.60



Linux News p.04



Chrome Cult p.41



Book Review p.51



Tuxidermy p.56



Ubuntu Games p.64



The articles contained in this magazine are released under the Creative Commons Attribution-Share Alike 3.0 Unported license. This means you can adapt, copy, distribute and transmit the articles but only under the following conditions: you must attribute the work to the original author in some way (at least a name, email or URL) and to this magazine by name ('Full Circle Magazine') and the URL [www.fullcirclemagazine.org](http://www.fullcirclemagazine.org) (but not attribute the article(s) in any way that suggests that they endorse you or your use of the work). If you alter, transform, or build upon this work, you must distribute the resulting work under the same, similar or a compatible license.

**Full Circle magazine is entirely independent of Canonical, the sponsor of the Ubuntu projects, and the views and opinions in the magazine should in no way be assumed to have Canonical endorsement.**



## WELCOME TO ANOTHER ISSUE OF FULL CIRCLE.

One up, one down. Elmer has had to take the month off from LibreOffice, but Greg is back for a quick Python column. Greg's been pretty sick these days, so if you're a Python fan feel free to email him a get-well-soon message: [greg.gregwa@gmail.com](mailto:greg.gregwa@gmail.com). We have a double install this month - one article shows how you can install an absolute minimalistic Xubuntu install, and one article shows how you can install \*buntu alongside Windows 10. Even if your machine has the evil UEFI enabled in the BIOS.

While YouTube recently announced a dedicated streaming service for games, there's always been Twitch. Oscar uses his Ubuntu Games column this month to discuss how to broadcast to Twitch using the Open Broadcaster Software (OBS). I'd love to do something like this, but my upload speed is nothing short of dire. Of course, OBS isn't just for games; it can broadcast anything from your machine.

By the time you read this, the latest OTA update (7 in the case of my Meizu) for Ubuntu phones should have been unleashed. To coincide with that, Lucas has dedicated his C&C this month (and next month) to coding for Ubuntu phones. The apps for Ubuntu phones were sparse, but they're getting better all the time. Two of my favourites have just been updated: **Activity Tracker** and **uNav**. Activity Tracker can track (via GPS) your walking, running or (in my case) cycling. It's not as sophisticated as some of its Android equivalents, but you can still go back and look at your route on the map. And Chris is always updating it. While uNav was always a route finder for cars, its developer (Marcos) has added features to now allow it to give route advice on cycle routes. I've not tried it yet, but will report back on it soon.

**All the best, and keep in touch!**

Ronnie

[ronnie@fullcirclemagazine.org](mailto:ronnie@fullcirclemagazine.org)



This magazine was created using :



## Find Full Circle on:



-  [goo.gl/FRTMI](https://goo.gl/FRTMI)
-  [facebook.com/fullcirclemagazine](https://facebook.com/fullcirclemagazine)
-  [twitter.com/#!/fullcirclemag](https://twitter.com/#!/fullcirclemag)
-  <http://issuu.com/fullcirclemagazine>
-  <https://play.google.com/store/books/author?id=Ronnie+Tucker>
-  <http://www.magzter.com/publishers/Full-Circle>



## LINUX CREATOR EXPLAINS WHY A TRULY SECURE COMPUTING PLATFORM WILL NEVER EXIST

Speaking at LinuxCon 2015 last month, Linus Torvalds talked about security as something unattainable in a perfect sense, something he expanded on with BGR. He thinks, for example, it's meaningless to ask what computing platform today is the most secure.

The most secure platform, he offers in response, is something that's "not actually usable."

"Unplug the network cable and instantiate draconian measures for physical security," he said. "You'll make sure nobody can get in, but you'll also make sure that nobody actually wants to use the platform. And that may sound like an extreme case, but it's a very fundamental issue in security. You cannot look at security as something separate."

Torvalds says he's butted heads with the security community because they often make a "complete circus" about things and think about things in terms that are too black-and-white.

Source: <http://bgr.com/2015/09/25/linus-torvalds-quotes-interview-linux-security/>

Submitted by: Arnfried Walbrecht

## INTEL: INVENTEC CONFIRMED IT IS MAKING THE XIAOMI LINUX LAPTOP

Since 2006, Apple found it prudent to consistently ignore the x86 processors from Advanced Micro Devices (NASDAQ:AMD). Consequently, its imitator, Xiaomi, is also unlikely to use them in its first laptop product. Intel's huge R&D spending has made its x86 CPUs higher performing than AMD's best processors.

Inventec Appliance Corp. is

designing/assembling the Xiaomi Linux laptop in its China factory. Along with Foxconn, Inventec is one of the top assemblers for Intel-powered notebook computers in China. Microsoft (NASDAQ:MSFT) also hired Inventec to help Lenovo (OTCPK:LNVGY) and Acer come up with low-cost sub \$250 Windows 10 notebooks using Intel Atom Bay Trail-T processors.

There is therefore great probability that Inventec will also use a CPU from Intel for Xiaomi's first laptop computer. The decision to use Linux is easily explained by the fact that Microsoft will never allow its proprietary Windows 10 OS to be customized by Xiaomi.

Source: <http://seekingalpha.com/article/3535486-intel-inventec-confirmed-it-is-making-the-xiaomi-linux-laptop>

Submitted by: Arnfried Walbrecht

## HADOOP OPEN DATA PLATFORM MOVES UNDER LINUX FOUNDATION'S WING

Six months down the line from its creation, the Open Data Platform Hadoop initiative driven by Pivotal and Hortonworks has today unveiled new members, work on a core spec and reference implementation, plus a formal governance structure.

The initiative caused controversy at its launch in February because of its declared aim of defining a core set of open-source Apache technologies to speed adoption of Hadoop.

Opponents dismissed it as a marketing effort and argued that interoperability across projects is not a major issue.

In a move that could further grate with those not in the Open Data Platform camp, the initiative is also now being hosted at the Linux Foundation as a collaborative project.



Source:

<http://www.zdnet.com/article/had-oop-open-data-platform-moves-under-linux-foundations-wing/>

Submitted by: Arnfried Walbrecht

## BOTNET PREYING ON LINUX COMPUTERS DELIVERS POTENT DDoS ATTACKS

Security researchers have uncovered a network of infected Linux computers that's flooding gaming and education sites with as much as 150 gigabits per second of malicious traffic—enough in some cases to take the targets completely offline.

The XOR DDoS (or Xor.DDoS) botnet, as the distributed denial-of-service network has been dubbed, targets as many as 20 sites each day, according to an advisory published Tuesday by content delivery network Akamai Technologies. About 90 percent of the targets are located in Asia. In some cases, the IP address of the participating bot is spoofed in a way that makes the compromised

machines appear to be part of the network being targeted. That technique can make it harder for defenders to stop the attack.

*"In short: Xor.DDoS is a multi-platform, polymorphic malware for Linux OS, and its ultimate goal is to DDoS other machines," a separate writeup on the botnet explained. "The name Xor.DDoS stems from the heavy usage of XOR encryption in both malware and network communication to the C&Cs (command and control servers)."*

Source:

<http://arstechnica.com/security/2015/09/botnet-preying-on-linux-computers-delivers-potent-ddos-attacks/>

Submitted by: Arnfried Walbrecht

## LINUX FOUNDATION: OPEN SOURCE CODE WORTH \$5B

How much is open source code worth? The answer: \$5 billion, according to a newly released Linux Foundation report that aims to illustrate the estimated value of development costs saved by the code embedded in its Collaborative

Projects.

The report, "A \$5 Billion Value: Estimating the Total Development Cost of Linux Foundation's Collaborative Projects," found that the total lines of source code that are present in the Collaborative Projects are 115,013,302.

The time that would be needed to recreate the total effort of these projects was found to be 41,192.25 person years, meaning it would take 1,373 developers 30 years to recreate the code bases.

And the price tag for that is about \$5 billion, the report concludes.

Source:

[http://www.informationweek.com/software/operating-systems/linux-foundation-open-source-code-worth-\\$5b/a/d-id/1322432](http://www.informationweek.com/software/operating-systems/linux-foundation-open-source-code-worth-$5b/a/d-id/1322432)

Submitted by: Arnfried Walbrecht

## WHY AREN'T WE ARGUING MORE ABOUT MR ROBOT?

In episode 0 of Mr Robot, we're introduced to our hero

protagonist [Elliot], played by Rami Malek, a tech at the security firm AllSafe. We are also introduced to the show's Macbeth, [Tyrell Wellick], played by Martin Wallström. When these characters are introduced to each other, [Tyrell] notices [Elliot] is using the Gnome desktop on his work computer while [Tyrell] says he's, "actually on KDE myself. I know [Gnome] is supposed to be better, but you know what they say, old habits, they die hard."

While this short exchange would appear to most as two techies talking shop, this is a scene with a surprisingly deep interpretation. Back in the 90s, when I didn't care if kids stayed off my lawn or not, there was a great desktop environment war in the land of Linux. KDE was not free, it was claimed by the knights of GNU, and this resulted in the creation of the Gnome.

Source:

<https://hackaday.com/2015/10/02/why-arent-we-arguing-more-about-mr-robot/>

Submitted by: Arnfried Walbrecht



## ANNIVERSARY OF FIRST LINUX KERNEL RELEASE: A LOOK AT COLLABORATIVE VALUE

The Linux community often recognizes two anniversaries for Linux: August 25th is the day Linus Torvalds first posted that he was working on Linux and said “Hello, everybody out there...” and October 5th is the day he released the first kernel.

To mark the anniversary of the first kernel release in 1991, we look at some facts and consider the progress that has been made since that early version.

\* Version 0.01 of the Linux kernel had 10,239 lines of code (source: Wikipedia).

\* Version 4.1, released in July 2015, has more than 19 million lines of code (source: Phoronix).

The current Linux kernel is the result of one of the largest collaborative projects ever attempted. According to the “Who Writes Linux” Linux development report published in February of this year:

Nearly 12,000 developers from more than 1,200 companies have contributed to the Linux kernel since tracking began 10 years ago.

The rate of Linux development is unmatched. The average number of changes accepted into the kernel per hour is 7.71, which translates to 185 changes every day and nearly 1,300 per week.

In recent years, the powerful growth of the Linux kernel and resulting innovation has inspired others to adapt the principles, practices and methodologies that make Linux so successful to solve some of today’s most complex technology problems.

Source: <http://www.linux.com/news/featured-blogs/185-jennifer-cloer/857378-anniversary-of-first-linux-kernel-release-a-look-at-collaborative-value>

Submitted by: Arnfried Walbrecht

## LINUX KERNEL DEV SARAH SHARP QUILTS, CITING ‘BRUTAL’ COMMUNICATIONS

## STYLE

A prominent Linux kernel developer announced in a blog post that she would step down from her direct work in the kernel community, saying that the community values blunt honesty, often containing profane and personal attacks, above “basic human decency.”

Sarah Sharp, an Intel employee who, until recently, was the maintainer of the USB 3.0 host controller driver, wrote that she could no longer work within a developer culture that required overworked maintainers to be rude and brusque in order to get the job done. She continues to work on other open-source software projects, but says that she has begun to dread even minor interaction with the kernel community.

Source: <http://www.networkworld.com/article/2988850/opensource-subnet/linux-kernel-dev-sarah-sharp-quits-citing-brutal-communications-style.html>

Submitted by: Arnfried Walbrecht

## OPEN NETWORK LINUX SIMPLIFIES OPEN COMPUTE PROJECT SWITCH CONFIGURATION

Big Switch Networks, Facebook and NTT have announced that they have come together to create a unified operating system called Open Network Linux for Open Compute Project’s (OCP) switch hardware.

While the name doesn’t exactly roll off the tongue, the project is designed to help companies, whether web scale-type companies like Facebook or others looking to take advantage of the Open Compute Project’s open source switches, to use the platform as a base to configure the switch’s forwarding algorithms (more on that in a minute) in a way that makes sense to them.

Up until now, the project has consisted of a set of disparate components that engineers had to stitch together. Open Network Linux helps bring these components together in a flexible way, while removing some of the

engineering complexity.

Source:

<http://techcrunch.com/2015/10/07/open-network-linux-is-ready-to-power-open-compute-project-switches/>

Submitted by: Arnfried Walbrecht

## REAL-TIME LINUX GETS A LEG-UP INTO MORE COMPLEX COMPUTING SYSTEMS

Back in 2006, Linus Torvalds said, "Controlling a laser with Linux is crazy, but everyone in this room is crazy in his own way. So if you want to use Linux to control an industrial welding laser, I have no problem with your using PREEMPT\_RT." The debate was started on whether Linux should be a real-time operating system.

Real-time Linux started years earlier when academics created the first real-time Linux distros such as eKURT, University of Kansas; RTAI, University of Milano; and RTLinux, New Mexico Institute of Mining and Technology. As the years went by, PREEMPT-RT, which

is maintained by Steven Rostedt, a Red Hat principal software programmer, became the most important real-time Linux variant. Disagreements on how to implement real-time functionality into Linux still exist. So The Linux Foundation, the non-profit organization dedicated to accelerating the growth of Linux and collaborative development, and its allies, created the new Real-Time Linux (RTL) Collaborative Project.

Source:

<http://www.zdnet.com/article/new-real-time-linux-project-launched-real-time-linux-rtl-collaborative-project/>

Submitted by: Arnfried Walbrecht

## UBUNTU FOR ANIME AND MANGA MANGAKA LINUX CHU SWITCHES TO CINNAMON AND GNOME

Celebrating seven years of activity and in the good tradition of the project to move to a different desktop environment

for each new release of the Mangaka Linux distribution, we report that the Mangaka Linux Chu OS will ship with a beautiful interface that combines elements from the popular Cinnamon and GNOME desktops. It will also include some of the latest and most popular Linux apps.

*"Now, our new team wanted to celebrate the 7 years of Animesoft International, releasing the Release Candidate of the new CHU that has Cinnamon+Gnome as desktop and filled with most recent Kodi media centre, Skype, Google apps, OpenOffice, Mozilla apps, Wine, codecs and multimedia editors installed out-of-the-box just for you!"* says Animesoft International in an email to Softpedia.

As you might know, Mangaka Linux is an Ubuntu-based computer operating system targeted at anime and manga fans, as it includes several applications for fansubbing and fandubbing. The final release of Mangaka Linux Chu will be available in the coming weeks and it'll be based on the latest Ubuntu 14.04 LTS (Trusty Tahr) release.

Source:

<http://news.softpedia.com/news/ubuntu-for-anime-and-manga-mangaka-linux-chu-switches-to-cinnamon-and-gnome-494322.shtml>

Submitted by: Arnfried Walbrecht

## THE LINUX FOUNDATION: HOW TO FIX THE INTERNET

The Linux Foundation, the organisation designed to promote Linux and open source software development practices, plans to improve internet security by coordinating teams of dedicated coders, and large firms with the financial power to fund them.

Speaking at technology conference IP EXPO in London today, Jim Zemlin, executive director of the Linux Foundation, began by outlining the ubiquity of Linux, the open source operating system originally developed by Linus Torvalds.

Since 2005, over 8,000 developers from around 800 firms have contributed to the Linux kernel (the fundamental part of

the operating system that translates user or other types of requests into instructions for the device's CPU). Zemlin said that a major new kernel comes out every two to three months, which is a far more regular update than other operating systems, like Microsoft's Windows platform, which usually sees new revisions only every five or more years.

The development process for Linux is extremely collaborative, and Zemlin highlighted this as a major strength of open source software in general, which he said firms are now seeking to turn to their advantage.

Source:  
<http://news.softpedia.com/news/ubuntu-for-anime-and-manga-mangaka-linux-chu-switches-to-cinnamon-and-gnome-494322.shtml>

Submitted by: Arnfried Walbrecht

## A DECADE OF LINUX PATENT NON-AGGRESSION: THE OPEN INVENTION NETWORK

**B**ack in 2005, Linux was still under attack by SCO for imaginary copyright violations, and Microsoft CEO Steve Ballmer was claiming that Linux violated more than 200 of the company's patents. Linux needed all the intellectual property (IP) law help it could get. So IBM, Sony, Philips, Red Hat, and Novell formed the Open Invention Network (OIN) patent consortium, to defend Linux against IP attacks. OIN's plan was to acquire Linux-related patents and share them royalty-free to any organization that agrees not to assert its patents against Linux or its applications.

It worked.

SCO is history. True, Microsoft, while embracing Linux and open source, is also still profiting from licensing never proved patents to Android vendors, but they're no longer rattling their legal sabers at the Linux distributors or Google.

Still, while Linux has IP legal fights on its hands, OIN has been a success story.

Source:  
<http://www.zdnet.com/article/a-decade-of-linux-patent-non-aggression-the-open-invention-network/>

Submitted by: Arnfried Walbrecht

## LINUX FOUNDATION AND ONOS PARTNER ON OPEN SOURCE SDN AND NFV NETWORKS

**O**NOS develops an SDN operating system for carrier-grade networks. Designed for high availability, high scalability and high performance, the platform is funded and supported by a range of industry partners, including AT&T, NTT Communications, SK Telecom, China Unicom, Ciena, Cisco, Ericsson, Fujitsu, Huawei, Intel and NEC.

The ONOS platform was open sourced in December 2014, and has issued four new releases since then.

As part of the partnership with the Linux Foundation, ONOS will "transform service providers'

infrastructure for increased monetization by achieving high capex and opex efficiencies and creating new innovative services using the power of open source SDN and NFV," the Linux Foundation said in a statement. "The Linux Foundation will assist ONOS to organize, grow and harness the power of this global community to take ONOS and the solutions enabled by it to the next level of production readiness and drive adoption in production networks."

Source:  
<http://thevarguy.com/open-source-application-software-companies/101315/linux-foundation-and-onos>

Submitted by: Arnfried Walbrecht

## LINUS TORVALDS IS "REALLY HAPPY" WITH LINUX KERNEL 4.3 RELEASE CANDIDATE 6

**L**inus Torvalds announced that the sixth Release Candidate of Linux kernel 4.3 is available for download and testing from the usual places, and it appears that things are calming down very well



for this release, which makes Mr. Torvalds really happy.

*"Things continue to be calm, and in fact have gotten progressively calmer. All of which makes me really happy, although my suspicious nature looks for things to blame," says Linus Torvalds. "Are people just on their best behavior because the Kernel Summit is imminent, and everybody is putting their best foot forward?"*

According to Linus Torvalds, Linux kernel 4.3 Release Candidate 6 consists of a great number of driver updates, especially for things like InfiniBand, which includes a copyright message clarification, and GPU (Graphics Processing Unit), various small architecture updates – mostly for x86 KVM (Kernel Virtual Machine) fixes for SMM emulation – as well as a few mm improvements.

Source:  
<http://news.softpedia.com/news/linus-torvalds-is-really-happy-with-linux-kernel-4-3-release-candidate-6-494775.shtml>

Submitted by: Arnfried Walbrecht

## US NUKE BOFFINRY TO BE POWERED BY FACEBOOK-INSPIRED LINUX SERVERS

Linux clusters built from Facebook's blueprints will help crunch numbers for the US government's hydrogen bomb scientists.

The computer system, dubbed the Tundra Extreme Scale series, will cost \$39m, and at its peak perform between seven and nine thousand trillion math calculations per second – that's seven to nine petaflops.

The machines will be installed at America's Los Alamos, Sandia, and Lawrence Livermore national laboratories from April 2016, with the last rack scheduled to be in place by September 2018. There, they will carry out "stockpile stewardship," which is a wonderfully sterile and bureaucratic way of saying nuclear weapon reliability testing and simulation.

Essentially, the computer system will be used to calculate whether or not Uncle Sam's stockpile of nukes, stored away in

grim silence, can be relied upon to wipe cities from the face of the Earth at short notice. Discovering your thermonuclear warheads have deteriorated into duds only after you press the big red button will be a bit of a bother. Politicians and military commanders want to avoid that scenario.

Source:  
[http://www.theregister.co.uk/2015/10/22/us\\_nuke\\_boffins\\_powered\\_by\\_ocp/](http://www.theregister.co.uk/2015/10/22/us_nuke_boffins_powered_by_ocp/)

Submitted by: Arnfried Walbrecht

## NTP FLAW IN LINUX, MAC, AND BSD OS DISTROS CAN BE USED TO COMPROMISE ENCRYPTION

Eight security vulnerabilities have been discovered by Cisco researchers in the Network Time Protocol (NTP) used by Linux, Mac, and BSD OS distributions. Network Time Protocol (NTP) is a networking protocol for clock synchronization between computer systems over packet-switched, variable-latency data networks. In operation since

before 1985, NTP is one of the oldest Internet protocols in current use. NTP was originally designed by David L. Mills of the University of Delaware, who still oversees its development.

One of the 8 security vulnerabilities discovered by Cisco's engineers allows attackers to manipulate a target's clock, making the victim believe they traveled to the future. Cisco engineers have stated that the vulnerabilities affect the Network Time Protocol daemon (ntpd), responsible for synchronizing time across computer networks (like the Internet, Intranets or smaller LANs).

Source:  
<http://www.techworm.net/2015/10/ntp-flaw-linux-mac-bsd-os-distros-compromise-encryption.html>

Submitted by: Arnfried Walbrecht



I recently received a BQ Aquaris E4.5 device running Ubuntu. Part of the reason for getting it was a promise to write a series of articles about developing for the phone. This will be part 1, covering setup and a basic 'hello world' program.

## INSTALL THE SDK

Ubuntu released the Ubuntu SDK in order to make development easier. I will be focusing on this program for the series. If you're averse to SDKs, I would imagine it's possible to still develop for Ubuntu, but I won't be covering it in this series.

## ADDING THE REPOSITORY

```
sudo add-apt-repository
ppa:ubuntu-sdk-team/ppa
```

This command will add the official ubuntu-sdk PPA, to allow you to easily get the newest packages.

## INSTALL THE PACKAGE

```
sudo apt update && sudo apt
install ubuntu-sdk
```

This command updates the package list, and immediately afterwards installs the ubuntu-sdk package.

The official install page notes that anyone running a development version of 15.10 should ensure their packages are all up-to-date, before installing ubuntu-sdk, with a "sudo apt dist-upgrade".

## LAUNCH THE APPLICATION

```
ubuntu-sdk
```

Or, just find and click on the sdk icon.

Once the application has launched, you'll want to click on "Create a New Project", or File -> New File or Project. For the sake of the Hello World program, I've chosen an HTML5 App. If you want to experiment with anything else, feel free to do so.

## CREATE YOUR PROJECT

The SDK will first ask you for a name and a save location. I chose HelloWorld and ~/Ubuntu SDK Projects/, but you're welcome to choose anything you want.

The next page will ask for personal information - your nickname, full name, email, the app name, and the framework you want to develop for. I chose the ubuntu-sdk-15.04 framework (as I am running 15.04).

## BUILD TARGETS

By default, there is only the Desktop kit available. So you'll need to choose "create new kit", and choose "armhf" if you're planning to run it on a phone. Once you select the architecture, it will prompt you for your password, and begin installing the kit. Note: If you're running this in a virtual machine (for any reason), or a small partition, make sure you have more than 10GB dedicated to the hard drive. (Using the Ubuntu base,

updates, the SDK, and the kit install, I ran out of space on my test 10GB partition).

Note: if you don't want an emulator, and want to run it only on a physical device, you can skip this step.

## POST KIT CREATION

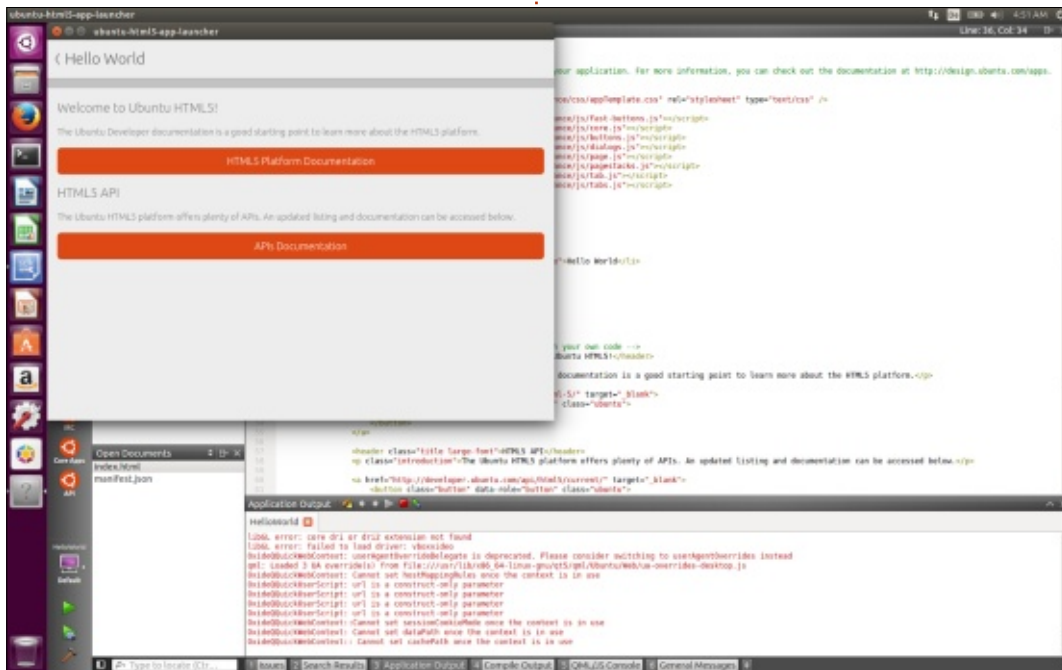
The wizard then simply asks what Kit(s) you want to use, and asks about version control. I enabled both Desktop and the armhf kit I just created, and skipped over the version control.

## THE HELLO WORLD APPLICATION

As it turns out, the basic HTML5 app is already a Hello World application. As such, we won't be doing any actual coding. Instead, I will focus on running the device locally, and on the physical device. If you want to adjust the HTML, feel free.

## DESKTOP

If you look in the lower left of the SDK window (shown below), you'll see an image of a Desktop (or an Ubuntu logo, depending on which kit you selected). Below that are two green arrows, and a hammer. The first green arrow is "run", the other is "debug", and the hammer is "build". The Desktop kit is used to run the application within Ubuntu, in a separate window. So, if you hit the "run" button, you should see something similar to the below screenshot (assuming you used an HTML5 App).



## UBUNTU PHONE (EMULATED)

In order to run it on an Ubuntu Phone, you'll need to create an emulator. To do so, go to Devices, and click the large plus sign. There, you'll need to give it a name (without spaces), choose the architecture, as well as the Ubuntu image you want. I chose bq-stable over devel, since my goal is to test it on a physical bq device later.

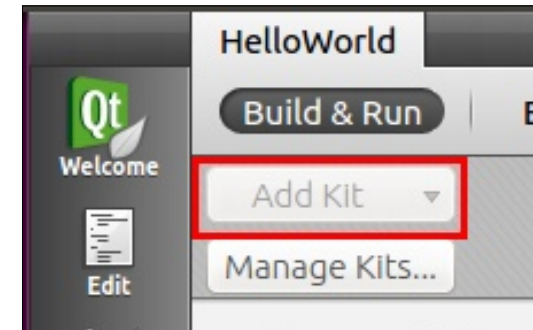
Once you enter the information, it will prompt you for a password, and then begin creating the emulator.

Once the emulator is created, simply click on the Icon above the green arrow, and select (with your arrow keys) the kit you want to use. In the case of the example, it's the armhf kit you want. Then clicking "run" will launch the emulator, and eventually load the app into the emulator.

## UBUNTU PHONE (PHYSICAL)

The Ubuntu Devices section from the emulated section will also display any usb connected Ubuntu devices. However, my Aquaris E4.5 was not recognized at first. It turns out I had forgotten to enable developer mode. On the phone, go to Settings -> About This Phone, and then Developer Mode. Make sure there is a green checkmark. Then plug it in. You can check to see if it's connected properly by running `adb devices` in the list. Or `lsusb` (to see if the device is even recognized). It should also appear as an MTP device. Once the physical device is connected, switch to the Ubuntu Devices panel, and ensure it's listed here too. Also, make sure it's unlocked. If it isn't, you'll see something similar to this in your log: **arch:error:closed**.

Once it appears in your Ubuntu Devices panel, it will show a message about frameworks. Just allow it to automatically select a framework, and you're done. This will create a new Kit called "Ubuntu Device". In order to build and run your application on the phone, you need to click on the tab/icon for "Projects". There, you'll want to click on the button "Add Kit" (just above the hammer icon), and select the Ubuntu Device from the menu.



Once the kit is added, you must select it (the icon above the green arrow in the bottom left), and run the project. This will then open the app on your phone.

In order to stop the application, click

on “Application Output” on the bottom of the window. There, you can click on a red square to stop the application from running. If you forget to do so, the Ubuntu SDK will warn you about it, and offer to force-quit the application.

## 3 Application Output

### WRAP-UP

That was it for this month. Now that we’ve gotten the basics out of the way (running applications), we’ll be ready to start programming! If you have any questions, requests, suggestions, or just want to say hi, I can be reached at [lswest34+fc@gmail.com](mailto:lswest34+fc@gmail.com).



**Lucas** has learned all he knows from repeatedly breaking his system, then having no other option but to discover how to fix it. You can email Lucas at: [lswest34@gmail.com](mailto:lswest34@gmail.com).



The Ubuntu Podcast covers all the latest news and issues facing Ubuntu Linux users and Free Software fans in general. The show appeals to the newest user and the oldest coder. Our discussions cover the development of Ubuntu but aren't overly technical. We are lucky enough to have some great guests on the show, telling us first hand about the latest exciting developments they are working on, in a way that we can all understand! We also talk about the Ubuntu community and what it gets up to.

The show is presented by members of the UK's Ubuntu Linux community. Because it is covered by the Ubuntu Code of Conduct it is suitable for all.

The show is broadcast live every fortnight on a Tuesday evening (British time) and is available for download the following day.

[podcast.ubuntu-uk.org](http://podcast.ubuntu-uk.org)



Welcome fellow pythoners. As the kids here in the central parts of the U.S. say, "What's Shakin' Bacon?" I'm not exactly sure what that's supposed to mean, but I assume it's a good thing.

You might notice the new header. I decided that I've taught you all the basics of Python that I can for "general" programming, so now we are going to delve into using Python to talk to other types of computers and controllers, like the Raspberry Pi and the Arduino micro controller. We'll look at things like temperature sensors, controlling motors, flashing LEDs and more.

This issue we will be focusing on what we'll need to do this and focus on a few of the projects we will be looking at in the future. Next issue, we will start the first project.

One of the things we will talk about next time will be the Raspberry Pi. The Pi is a credit-card sized computer that natively runs

Linux on an SD card. Its output goes to your TV set via HDMI. It also has an Ethernet connection for Internet access.

You can find out more at the official site <https://www.raspberrypi.org>. If you want to follow along with the projects, you will need a Pi, SD card, Keyboard, Mouse, a 5volt DC power supply like the ones on modern cell phones, and access to an HDMI monitor or TV. Eventually, you should also consider getting a breadboard and some connecting wires for when we start to interface to the outside world. You can find any number of places that sell the Pi on the Internet. Here in the U.S., we can get them for around \$35.

One other thing about the Pi is that it provides access to a series of pins that support GPIO (General Purpose Input/Output). Basically, this means that you can write programs that will send signals to the output pins and read the signals from the input pins. This can be used to interface to things

like LEDs, sensors, push buttons, etc. Many people have made home automation systems, multiple processor systems (by linking 40 or so Pi computers together to emulate a supercomputer), weather stations, even drones. So you can imagine that the possibilities are endless. That's why I decided to start with it for this series of articles.

After a while, we will begin to work with the Arduino, which according to the official website (<https://www.arduino.cc>): "*Arduino is an open-source electronics platform based on easy-to-use hardware and software. It's intended for anyone making interactive projects*".

Once again, this is an exciting device to work with. In this part of the series, we will look at talking to the Arduino, first in its native scripting language, and then in Python and eventually interfacing the Pi with the Arduino.

I know this month's article is fairly short, but I've been doing

poorly health-wise, so I'm saving my strength for the next article. Until then, grab some electronics and get ready for fun!



**Greg Walters** is owner of RainyDay Solutions, LLC, a consulting company in Aurora, Colorado, and has been programming since 1972. He enjoys cooking, hiking, music, and spending time with his family. His website is [www.thedesignatedgeek.net](http://www.thedesignatedgeek.net).



I have a PC with a hard disk of 1TB, a solid state drive of 250 GB, and a CD/DVD drive. I want to install Windows 10 and four Linux distros on it using UEFI and secure boot.

Previously, I had a lot of problems when I made a multi-boot system with Linux distros, because the boot loader (grub2) and its directory (/boot/) must be shared. The problem is that each distro installs a different version of grub2. The result could be that the PC refuses to boot after a distro update due to inconsistencies between the updates that the distro made in the /boot directory and the boot loader installed on the hard disk.

To prevent this I intend to use the two-step boot mechanism described in article "Howto: GRUB2 and multiboot Pt. 4" that was published FCM#88. The central boot loader gets a separate partition for its data and it will allow the user to select a distro from a menu. The central boot loader will then start the specific

boot loader of the selected distro. The specific boot loader has its own partition for its data, and it will show a second menu to the user to select a particular kernel version and eventually extra boot options. The chainloader module of grub2 is used to start another boot loader.

I also use LVM (Logical Volume Manager) as much as possible. In LVM, one can assign one or more physical partitions to a so-called volume group, and create logical partitions in the volume group. These logical partitions are called logical volumes in LVM speak. It is very easy to change the size of logical volumes when needed. Another advantage is that the name of a logical volume can not change, so you can safely address a logical volume lv in volume group hdvg as /dev/mapper/hdvg-lv or /dev/hdvg/lv.

I intend to create one volume group "hdvg" on the hard disk for the volatile data of the distros, and another one "ssdvg" on the solid state drive. I will install Windows

10 on the first half of the hard disk.

The central boot loader needs a physical partition (the boot loader does not speak LVM), and I will place it on the solid state drive. Other partitions that are shared between all distros are:

- a physical partition used as swap area, and
- a logical volume for temporary files that are erased when Linux starts up (/tmp).

Both partitions will be on the hard drive as they contain volatile data.

I will create for each distro:

- a physical partition on the solid state drive for the specific boot loader (/boot),
- a logical volume on the solid state drive for the root file system of the distro (/), and finally
- a logical volume for persistent data (/var) on the hard disk.

One usually also provides a separate partition for the user data (/home), but I will not do that because I have a NAS from which I

mount my home directories in the preferred distro (Kubuntu). The other distros are just for experimenting, so they never contain important data.

## STEP 1: PREPARE FOR THE INSTALLATION OF WINDOWS 10

Decouple first the solid state disk from the motherboard to make sure Windows 10 uses only the hard disk.

Windows 10 is sold in the form of a USB stick, so plug the stick in one of the USB slots and boot the PC. Enter the BIOS to enable UEFI and secure boot. For my motherboard type, I must press the Del key a few times just after the PC boots.

## STEP 2: PREPARE THE BIOS

The nasty thing is that the procedure depends on the motherboard type, so you probably have to experiment to find the exact settings for your



motherboard. Mine is an ASUS A88X-Plus. For this type of motherboard, one must enter Advanced Mode and then select the Boot tab.

To enable UEFI, go into "Compatibility Support Mode" and set:

- "Launch CSM" to Enabled,
- "Boot Device Control" to "UEFI only",
- "Boot from storage devices" to "Both, UEFI first" and
- "Boot from PCIe/PCI expansion devices" to "UEFI drive first".

To enable secure boot, set "Secure boot" to "Windows UEFI mode".

Make the USB stick the first boot device. I had to select "UEFI: KDI-MSFTWindows10". Always select the UEFI variant in case you have multiple options. Save the configuration and exit.

## STEP 3: INSTALL WINDOWS

The PC reboots now and starts with the installation of Windows 10.

The installation program

proposes to divide the hard disk in four partitions. I reduced the size of the largest partition to 500 GByte so I have enough space for the future Linux partitions.

During the installation, you have to answer a few questions, but finally your PC reboots into Windows 10.

In a multiboot environment, you must disable fast reboot to avoid corruption of your Windows 8, 8.1 or 10 installation.

Go to the "Control Panel", select System, "Power options," and then "Choose what the power options do". Click on "Change settings that are currently unavailable," and finally remove the tick in front of "Turn on fast startup (recommended)". "Save changes" and reboot your PC.

Now remove the USB stick with the Windows 10 software.

## STEP 4: INSTALL KUBUNTU

Download the iso-image of Kubuntu 14.04 LTS 64-bit, burn it on a DVD, put it in the CD/DVD drive, and shutdown the system.

Reconnect the solid state drive to the motherboard and boot the PC.

Go into BIOS and select the DVD as boot device. I had to select "UEFI: P3 TSTST corp CDDVDW SH-224 DB". Always select the UEFI variant in case you have multiple options. Save the configuration and exit.

Select first "Start Kubuntu", wait some time until you can select your language, and "Install Kubuntu". In step "Disk Setup," choose "Manual" as "Installation Type".

Bummer: the installer does not allow to create LVM volume groups and logical volumes, but it can work with logical volumes if they are created in advance.

So you now have the choice:

- forget about LVM and create only physical partitions, or
- jump to the Kubuntu live and create the physical partitions and logical volumes there.

I will explore the latter option.

## STEP 5 : CREATE PARTITIONS

## AND LOGICAL VOLUMES IN KUBUNTU LIVE

Quit the installation, and Kubuntu live is started automatically.

Open now a terminal: you find one under Applications->System as "Terminal Konsole".

Install gparted with following command:

```
sudo apt-get install gparted
```

Start gparted:

```
sudo gparted
```

Select the appropriate disk: you should be able to figure out from the sizes which one is the hard disk and which one is the solid state drive. Make sure to remember the identification (for example: /dev/sda7) of each partition you have created.

The hard disk already has a partition table. Select the unallocated area and select Partition->New to add a new partition: give it a size of 1GB and select linux-swaps as file system.

This partition will be used as swap space. Repeat the process to add another partition, with file system lvm2pv, that spans the complete unallocated area. Press the Apply button.

Note: there is no need to format the partitions: let the installer do that.

The SSD drive does not have a partition table yet. Select the device and select Device->Create Partition Table. Under Advanced, change the type from msdos to gpt, and Apply.

Create now 6 partitions:

- one of 100 Mbyte with file system ext2 for the central boot loader,
- four of 1 GByte also with file system ext2 for the specific boot loaders, and
- a last one with file system lvm2pv that spans the complete unallocated area.

Press the Apply button. Close gparted and return to the terminal.

Add the physical partition with file system lvm2pv that we just created on the hard disk (in my case this was called /dev/sdb6) to LVM:

```
sudo pvcreate /dev/sdb6
```

Create the volume group hdvg and assign the partition to it:

```
sudo vgcreate hdvg /dev/sdb6
```

Create logical volume var1 of 30 Gbyte in volume group hdvg:

```
sudo lvcreate -n var1 -L 30G hdvg
```

In the same way, create logical volumes var2, var3 and var4 and a 10 Gbyte logical volume tmp.

Create now the volume group ssdvg on the solid state drive, and create four logical volumes root1, root2, root3, root4 of 20 GByte each.

Go back to the installer by pressing "Install Kubuntu" on the desktop.

## STEP 6 : RESTART THE KUBUNTU INSTALLATION

Choose again, "Manual" as "Installation Type" in step "Disk Setup".

Use the central boot partition

for /boot/central, and the first specific boot partition for /boot. In both cases, select ext2 as the file system.

Use /dev/ssdvg/root1 as /, /dev/hdvg/var1 as /var, and /dev/hdvg/tmp as /tmp, all with a ext4 file system. Finally use the swap partition as swap space. Let the installer format all partitions.

Continue with the installation. Install the boot loader on the solid state drive and finally your PC will reboot.

If you boot directly into Windows, enter the BIOS and make sure that Ubuntu is the first boot device. I had to select "Ubuntu (P4 : WDC WD10EAVS-00D7B1)".

You should now be able to start up Windows and Kubuntu via the grub menu.

## STEP 7 : CONFIGURE LINUX TO USE THE SSD

Note: in this step you will make some changes to configuration files. Be very careful doing so: always make a backup file of the original version (sudo cp config-file

config-file.bak), and add a comment with your name and date, and the reason why you made the change to the modified file. In this way you can easily revert changes or find all files that you modified. Use the live DVD to repair in case things go terribly wrong.

You must reduce as much as possible the number of writes to your solid state drive to increase its lifetime.

The file system, by default, writes the access time of each file or directory that you read. This information is almost never used, so it is safe to disable this feature. You should at least do this for the solid state drive, but it does no harm if you do this too for your hard disk as it will make your disk access faster.

Edit /etc/fstab as root, and add "noatime" in the options field (don't forget the comma) of the partitions or logical volumes that are on any of the disks.

Example:

```
UUID=8482863b-d04e-40d2-be10-f5f3df88b8cd / ext4
```



```
errors=remount-ro 0 1
```

```
UUID=f65f89ac-b2b0-4345-949a-6965e3513db3 /boot ext2
defaults 0 2
```

becomes:

```
UUID=8482863b-d04e-40d2-be10-f5f3df88b8cd / ext4
errors=remount-ro,noatime 0 1
```

```
UUID=f65f89ac-b2b0-4345-949a-6965e3513db3 /boot ext2
defaults,noatime 0 2
```

Verify that you entered the correct syntax by executing: `sudo mount -a`.

A second modification is to run the trim command at boot up time and not via a cron job. Edit `/etc/rc.local` as root, and add “`fstrim -v <partition>`” for each partition that is written on the solid state drive that is written often by this distro. I have added:

```
fstrim -v /
```

Edit `/etc/cron-weekly/fstrim` as root, and put a hash ‘#’ in front of “`fstrim-all`”. Example:

```
#exec fstrim-all
```

In case your cache partition is on the solid state drive, you should

also reduce the number of times Linux uses the cache by editing `/etc/sysctl.conf` as root and adding following lines:

```
vm.swappiness=1
```

```
vm.vfs_cache_pressure=50
```

Some applications like Firefox and Java write a lot to the home directory. Also this causes unnecessary wear of the solid state drive. For example, google “firefox and ssd” to find instructions to make applications solid-state-drive friendly.

## STEP 8 : CONFIGURE THE 2-STEP BOOT MENU

I found my inspiration in article “Howto: GRUB2 and multiboot Pt. 4” that was published in FullCircle 88. Our central boot partition, `/boot/central`, is the equivalent of `/mnt/GRUBpart/boot` in the article. Take into account that UEFI requires a lot of changes to the procedure.

First, add chain loader entries to `/etc/grub.d/40_custom`.

Chain loader entries for use

```
menuentry 'Kubuntu 14.04 amd64 op /dev/sda9' {
    insmod part_gpt
    insmod chain
    set root='hd0,gpt5'
    set prefix=($root)/grub
    configfile $prefix/grub.cfg
    set efi_root='hd2,gpt2'
    chainloader ($efi_root)/EFI/ubuntu/grubx64.efi
}
```

without UEFI are very simple. Such an entry looks as follows: if `hd0,msdos1` is the grub name of the partition where you want to jump to (the partition that is mounted on `/boot` or on `/` if you have no separate boot partition):

```
menuentry 'Ubuntu' {
    set root='hd0,msdos1'
    chainloader +1
}
```

**Hint:** inspect `/boot/grub/grub.cfg` to find out how partitions are named by grub. Use the grub shell in case you are in doubt about the correct names: reboot your PC, go into the BIOS, disable secure boot, save and exit, and press escape when you see the grub menu. You can now enter commands like `dir (hd0,msdos1)/` to see the contents of a given partition: this will help you to verify if an assumed grub name is correct. Use the command `reboot` to reboot the PC.

A chain loader entry for UEFI is much more complicated. It looks as shown above, if `hd0,gpt5` is the grub name of the partition where you want to jump to (the partition that is mounted on `/boot` or on `/` if you have no separate boot partition), and if `hd2,gpt2` is the grub name for the EFI partition.

**Note:** check and correct the paths to the different files when you use another distro and/or partition scheme!

It is probably better to work with the uuid as the hard disk numbers (`hdx`) in grub can change if you, for example, start up with an USB stick plugged in. Use `sudo blkid /dev/sda5` to find the uuid of partition `/dev/sda5`. The chain loader entry looks now as shown (page after next) at the bottom (don't forget to enter the correct values for the hints too).

# HOWTO - MULTIMODE UEFI

Add a chainloader entry (previous page, top right) to jump to Kubuntu itself and copy the entries for “Windows 10” and “System Setup” from /boot/grub/grub.cfg to /etc/grub.d/40\_custom.

Execute:

```
sudo update-grub
```

reboot, and verify that the new entries work. Correct if necessary.

Prepare the central boot directory:

```
sudo mkdir /boot/central/efi

sudo mount -o bind /boot/efi
/boot/central/efi

sudo cp
/boot/efi/EFI/ubuntu/grub.cfg
/boot/efi/EFI/ubuntu/grub.cfg
.bak

sudo grub-install /dev/sda
--boot-directory=/boot/central
```

Make a backup of grub.cfg, generate the grub.cfg first for the central boot loader, and afterwards again for Kubuntu.

```
sudo cp /boot/grub/grub.cfg
/boot/grub/grub.cfg.bak
```

```
cd /etc/grub.d

sudo chmod -x *linux* *mem*
*prober* *uefi*

sudo update-grub

sudo cp /boot/grub/grub.cfg
/boot/central/grub/

sudo cp
/boot/grub/unicode.pf2
/boot/central/grub/

sudo chmod +x *linux* *mem*

sudo chmod -x *custom*

sudo update-grub
```

Prepare the central boot loader and create 2 new boot entries, called central\_grub and

central\_shim. The -d option indicates the disk that contains the efi partition, and -p is the number of the efi partition. The efi partition in my case is /dev/sdb2 so we get:

```
sudo cp -R
/boot/efi/EFI/ubuntu
/boot/efi/EFI/central

sudo mv
/boot/efi/EFI/ubuntu/grub.cfg
.bak
/boot/efi/EFI/ubuntu/grub.cfg

sudo rm
/boot/efi/EFI/central/grub.cf
g.bak
```

```
menuentry 'Kubuntu 14.04 amd64 op /dev/sda9' {
  insmod part_gpt
  insmod chain
  set root='hd0,gpt5'
  if [ x$feature_platform_search_hint = xy ]; then
    search --no-floppy --fs-uuid --set=root --hint-bios=hd0,gpt5 --hint-efi=hd0,gpt5 --hint-baremetal=ahci0,gpt5
f65f89ac-b2b0-4345-949a-6965e3513db3
  else
    search --no-floppy --fs-uuid --set=root f65f89ac-b2b0-4345-949a-6965e3513db3
  fi
  set prefix=($root)/grub
  configfile $prefix/grub.cfg
  set efi_root='hd2,gpt2'
  if [ x$feature_platform_search_hint = xy ]; then
    search --no-floppy --fs-uuid --set=efi_root --hint-bios=hd2,gpt2 --hint-efi=hd2,gpt2 --hint-
baremetal=ahci2,gpt2 EC4E-2E34
  else
    search --no-floppy --fs-uuid --set=efi_root EC4E-2E34
  fi
  chainloader ($efi_root)/EFI/ubuntu/grubx64.efi
}
```

```
sudo efibootmgr -c -l
\\EFI\\central\\grubx64.efi
-L central_grub -d /dev/sdb
-p 2
```

```
sudo efibootmgr -c -l
\\EFI\\central\\shimx64.efi
-L central_shim -d /dev/sdb
-p 2
```

Verify the contents of /boot/efi/EFI/central/grub.cfg and /boot/efi/EFI/ubuntu/grub.cfg. These files have following content:

```
search.fs_uuid 5b686b70-7fdf-
495c-afa8-33847392b06f root
hd0,gpt1
```

```
set prefix=($root)'/grub'
configfile $prefix/grub.cfg
```

Make sure that uuid and root refer to, respectively, the central boot partition and the kubuntu specific boot partition. Correct if necessary.

## STEP 9: INSTALL THE OTHER DISTROS

The next distro I installed was Debian but I was obliged to disable secure boot in the BIOS first.

Installation is similar to the installation of the first distro, except that you don't have to

worry anymore about creating partitions. Make sure that /boot/central, /tmp, and the swap partition, are not formatted again.

The installers of some distros always format the swap partition. This partition will then get a new uuid. In that case you need to correct the uuid of the swap partition in the /etc/fstab file of the other distros.

Add now a chain loader entry in /boot/central/grub/grub.cfg for the new installed distro. Make sure you use the correct paths: for Debian you must replace ubuntu/grubx64.efi by debian/grubx64.efi.

Reboot, go in the BIOS, enable secure boot, and make central\_shim the default boot loader.

## THE END

You should now be able to easily switch between Kubuntu, Debian and Windows 10. Enjoy!

A last remark: you can not combine grub splash images with Secure Boot because the image

files are considered to be insecure by the BIOS as they are not signed.



**Frank** graduated as a civil engineer, mechanical in 1986 and now develops software for the big Internet routers. He uses Linux at home and professionally. He is very interested to find out how things like Linux work under the hood.





Now that our Linux VM is built and secure, it's time to install the web server.

## WHAT EXACTLY IS A WEB SERVER?

A web server is software which serves web pages (and potentially other files – for example binaries – like videos, packages, ... etc).

And how does this really actually work? The web server runs as a daemon. "Daemon" - at least in the \*nix family – means software that runs in background (meaning there is typically no output on the screen, the program runs silently without interaction from the user), and typically listens on a TCP port (more on TCP ports in part 2, published last week).

When a request is sent to that specific listening port, the daemon wakes up and produces an action – for a web server, the daemon typically sends back a web page.

## WHICH WEB SERVER TO

### CHOOSE?

The most popular web servers are Apache, nginx, Microsoft and Google (list taken from netcraft.com). I am not really familiar with Google's web server offering, and Microsoft is definitely off the list (try to wonder why!) - so we get to choose between Apache and nginx.

Apache has been around for longer and has the largest market share. Nginx is supposed to be lighter and therefore maybe faster.

I chose Apache for this tutorial – there is no specific reason why not nginx, which is also an excellent server, other than I personally have more exposure with Apache software in general.

### INSTALL APACHE WEB SERVER

Before we start, note that I will also add the commands for Centos7.

Installing a web server is as easy as running this command!

```
sudo apt-get install apache2
(Centos7 - yum install httpd)
```

Make sure to answer Y to continue (or hit enter)

We can now check that the web-server started properly (shown below) - using a web browser, key the IP address of the server (in my example it is 159.203.90.111).

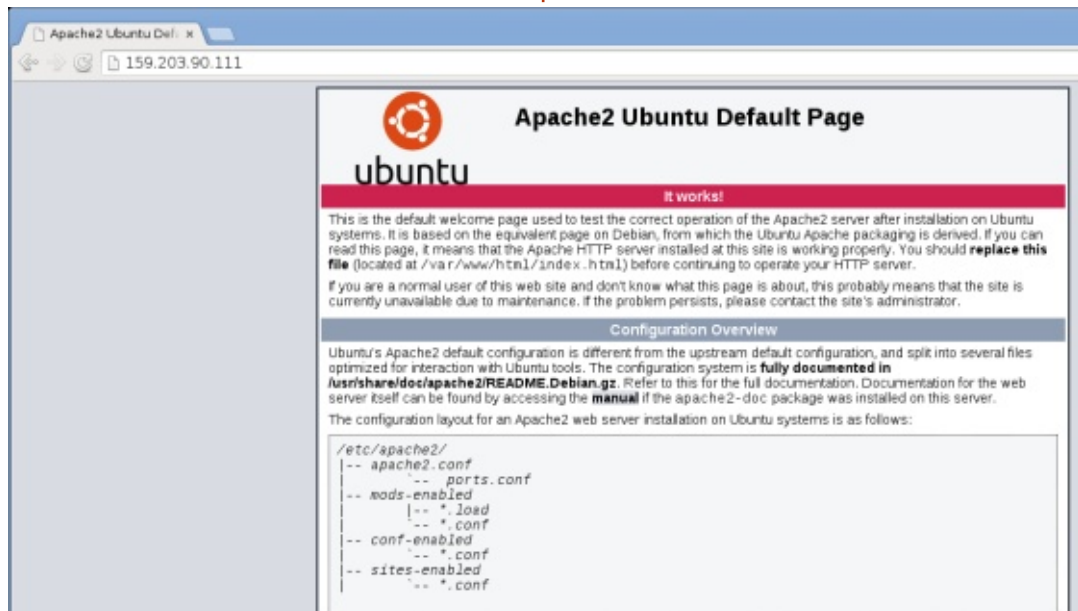
### CONFIGURATION AND TUNING

Now that Apache is installed, we must tune, configure and secure the web server.

### TUNING FIRST

Typically, tuning is done at the end. Having said this, I personally tend to forget about tuning and therefore I can end up with a web server that's sluggish – should workload pick up. So let's tune it right now – it won't have any side effect.

Edit the file



# HOWTO - WEBSITE WITH INFRASTRUCTURE

/etc/apache2/apache2.conf, and add the following at the end of the file (for Centos7, it's: /etc/httpd/conf/httpd.conf):

```
sudo vi /etc/apache2/apache2.conf
```

```
<IfModule mpm_prefork_module>
    StartServers 2
    MinSpareServers 6
    MaxSpareServers 12
    MaxClients 80
    MaxRequestsPerChild 3000
</IfModule>
```

For these to take effect, save the file and restart the Apache service with:

```
sudo service apache2 restart
(Centos7: systemctl restart httpd)
```

What does this all mean?

- StartServers defines the minimum number of child server processes created when web server starts. 2 works well for me, not sure what the default is.
- MinSpareServers is the minimum number of threads waiting for requests while MaxSpareServers is the maximum number. Higher the number, more load the server can handle, however we have to balance the values with our server resources (1 CPU & 512MB of

RAM). 6 and 12 work well here.

- MaxClients is the max number of simultaneous requests that will be served (any additional will be queued). 80 works well here.
- MaxRequestsPerChild is the threshold after which a child process will re-spawn. For example, as any software, Apache can have memory leaks – so restarting the child process after a given number of requests served will clean up potentially leaked resources.

## DISABLE DEFAULT SITE

It is now time to disable the default site, meaning the page which was served when we keyed in the IP address of the server. Basically we want to do this for security and convenience reasons - when somebody keys in the IP address of my server, I'd rather send the user to my web page than the default Apache page.

First we have to find the name of the default site

```
root@iceberg:/etc/apache2/sites-enabled#
root@iceberg:/etc/apache2/sites-enabled#
root@iceberg:/etc/apache2/sites-enabled# ls
000-default.conf
root@iceberg:/etc/apache2/sites-enabled#
```

To disable, use:

```
sudo a2dissite 000-default
```

Check the sites-enabled folder – it is now gone!

Restart the server (service apache2 restart) – basically no “site” served anymore, just a folder browser:



## CREATE OUR SITE

Our site will be iceberg-tutorial.com (iceberg.com is already taken!), so we will create a configuration file called iceberg-tutorial.conf (note: we could have chosen any name – I just assume that using a configuration filename with the same name as final site just helps in the long run for maintenance):

```
sudo vi /etc/apache2/sites-available/iceberg-tutorial.conf
```

And add all this to the file (in Centos7, the folder is: /etc/httpd/conf.d)

```
<VirtualHost *:80>
    ServerAdmin
your_email@here.com
    ServerName iceberg-tutorial.com
    ServerAlias *.iceberg-tutorial.com
    DocumentRoot
/var/www/iceberg-tutorial/public_html/
    ErrorLog
/var/www/iceberg-tutorial/logs/error.log
    CustomLog
/var/www/iceberg-tutorial/logs/access.log
    combined
</VirtualHost>
```

What does this all mean?

- Apache is listening on port 80 (more below).
- ServerName is the name of your website.
- DocumentRoot is the path where the files of the web server are stored.
- ErrorLog defines the path of where error logs are stored.

We have therefore to create the path to these folders:

```
sudo mkdir -p /var/www/iceberg-tutorial/public_html/
```

# HOWTO - WEBSITE WITH INFRASTRUCTURE

```
sudo mkdir -p
/var/www/iceberg-
tutorial/logs
```

And also make sure these folders and files can be read:

```
sudo chmod -R 755 /var/www
```

And finally enable the site:

```
sudo a2ensite iceberg-
tutorial.conf
```

If we try to access the website, we'll get this – this is expected behavior since there are no files (we created only the folders):



By default, Apache is looking for a file called index.html – let's create one:

```
sudo vi /var/www/iceberg-
tutorial/public_html/index.ht
ml
```

Key in, for example, 'Hello there!', then close and save.

Refresh the page, you should now see something like this:



## QUICK NOTES ABOUT TCP PORTS

We already spoke about TCP ports in the previous article. A great tool to check what ports are open is nmap. To scan the first 1000 ports, type:

```
sudo nmap localhost
```

and you will see which ports are open.

We can see here that 22 (SSH) and 80 (http) are open – which is expected.

To scan other port ranges, you can use the -p option (ex: nmap -p 2000-3000 localhost).

## VIRTUAL SITES

It is possible to host several sites on the same server. Since the server has a unique IP address, the originating URL will help Apache go to the right site - in other words, serve the pages from the correct folder.

So we can have several configuration files in the folder /etc/apache2/sites-available/

For example (remember that for maintenance reasons, the name of the folder is the name of the URL itself):

```
iceberg-tutorial.conf
... DocumentRoot
/var/www/iceberg-
tutorial/public_html/ ...
```

```
whatever-site.conf
... DocumentRoot
/var/www/whatever-
site/public_html/ ...
```

If the originating URL is www.iceberg-tutorial.com, Apache will serve the pages from /var/www/iceberg-tutorial/public\_html/, while, if the originating URL is www.whatever-site.com, Apache will serve the pages from /var/www/whatever-site/public\_html/

## SECURITY

What would a web server setup be without security? It would probably be like leaving your car in the garage with the keys on the ignition – somebody may steal the car or not. Maybe it's not a great analogy, but you probably got the point!

Apache is open source software, therefore it is very easy to add modules and there are a bunch of security modules available.

Remember however that security is not foolproof – it is only a mitigating factor – so you must pro-actively check the system logs for intrusions or attempts of intrusion. Let's make another analogy – it's like you purchased that outstanding vault. Breaking into that vault will be difficult, but if an attacker has enough time and the right skills, he could potentially break inside. Same here – check frequently your system logs (more in the howto article on that).

## OUT OF THE BOX SECURITY

By “out of the box” is meant

that no download is required – just add all below to the end of the file /etc/apache2/apache2.conf:

```
ServerTokens Prod
ServerSignature Off
```

```
FileETag None
TraceEnable off
Timeout 60
```

```
<Directory />
  Options None
  AllowOverride None
  Order deny,allow

  <LimitExcept GET POST
HEAD>
  deny from all
  </LimitExcept>
</Directory>
```

```
LoadModule headers_module
/usr/lib/apache2/modules/mod_
headers.so
```

```
Header edit Set-Cookie ^(.*)$
$1;HttpOnly;Secure
```

```
Header always append X-Frame-
Options SAMEORIGIN
```

Quick comments what all this means

- ServerTokens and ServerSignature will not disclose the Apache version (signature of web server will be just Apache). This avoids giving an attacker information regarding specific exploits available for your version (eg: just making this up – version

1.4.2 has the XXX known vulnerability).

- TraceEnabled doesn't allow debugging (additional trace) and Timeout is probably self explanatory!
- The Directory directive adds restrictions on root folder.
- As explained earlier, Apache can easily be enhanced with modules. Here we ask Apache to load the headers\_module, which will be used in the 2 commands below (Header Edits ... and Header always ...) in order to block XSS or using iFrames attacks. I am not an expert on these topics – my recommendation is copy-paste the entire line (Header Edits ... and Header always ...) into Google for more details.

## ADDITIONAL SECURITY – MODULE MODSECURITY FOR APACHE

Extremely popular for Apache servers (and maybe others), ModSecurity is a must-have module. Note this is not mandatory, however I highly recommend installing it (it's free and it adds security – so why not?).

Once ModSecurity is installed, it doesn't do anything out of the box – you must turn on what options you need. To make things easier, common rules (also called CRS - Core Set Rules) are available and just need to be turned on.

There are many websites with all the instructions on how to install ModSecurity and turn on CRS – for all step-by-step instructions, please follow the link below (there are many other tutorials available online on how-to proceed):

<https://www.digitalocean.com/community/tutorials/how-to-set-up-modsecurity-with-apache-on-ubuntu-14-04-and-debian-8>

## CLEANUP AND SOME STATISTICS

The folder /var/www/iceberg-tutorial/logs will start filling up with logs:

```
root@iceberg: /var/www/iceberg-tutorial/logs#
root@iceberg: /var/www/iceberg-tutorial/logs# ls -l
total 4
-rw-r--r-- 1 root root 2498 Oct  6 20:42 access.log
-rw-r--r-- 1 root root   0 Oct  5 21:44 error.log
root@iceberg: /var/www/iceberg-tutorial/logs#
```

Check out access.log – you should see the IP address from which you have accessed the web server (meaning the IP address of

where the browser ran) – cool stuff, isn't it?

We can now also run statistics – what pages were open, IP source, browser info, ... etc. Of course, you can use Google analytics; however as a pure geek, I personally enjoyed browsing the web logs to get my own stats.

Note that the log file size will continue increasing – so we must clean it up. This very small script (shown on the next page) will count all unique access to the web server and then compress the log. It's a bash job, all lines starting with # are comments. Once the script is created, you can add the script to a cron job to run daily:

## FINAL NOTE – WEBSITE REGISTRATION AND DNS

Once the web server is set up and pages ready, you will probably register a website name – it is easier to remember www.iceberg-tutorial.com rather than 159.203.90.111.

In order to do this, find your favorite website registrar online, and follow all the steps (you'll have

# HOWTO - WEBSITE WITH INFRASTRUCTURE

to pay something – usually it's around \$15 for one year).

You will then have to also set up a DNS entry at Digital Ocean – this is the link which controls where the browser will jump to – when anyone tries to access `www.iceberg-tutorial.com` (no additional charge – free at last!).

All information on these steps is very well explained here: <https://www.digitalocean.com/community/tutorials/how-to-set-up-a-host-name-with-digitalocean>

I hope you have enjoyed these articles and that you will create your own website, from scratch!

```
#!/bin/bash

cd /var/www/iceberg-tutorial/logs

# Get the current date in format YYYY-MM-DD
MYDATE=$(date +%Y-%m-%d)

#Get some stats

# Do not count any bot (grep -v bot)
# Do not count any internal IP V6 access ":::1"
# Do not count me grep -v 159.203.90.111
# Print 1st field (that is the IP addresses
# Sort all IPs
# Keep only uniq IPs

# Count them and add to file using >> (which means "append"). Caution > would mean
"overwrite"

# ==> The file statistics.txt will keep for each day the total number of unique IP
addresses which have accessed the web site

RESULT=$(cat access.log | grep -v -i "bot" | grep -v ":::1" | grep -v 159.203.90.111 |
gawk '{print $1}' | sort | uniq | wc -l)

echo $MYDATE " " $RESULT >> statistics.txt

# Compress the log

cat access.log | xz > access__$MYDATE.xz

# Truncate the file (basically it is like rm file && touch file)

:>access.log
```





The best way to be secure on your computer is to control what is installed on it. For the most control, I would suggest starting off with an Ubuntu Minimal Install and build upon that.

I chose Ubuntu because it (and its derivatives... like LinuxMint) is (in my opinion) the most user-friendly Linux distro out there. It is very easy to get support from the forums; the repositories contain most of the useful software; new repositories/PPA's are easy to add; you can install via .deb files (akin to M\$ .exe/.msi files); you can easily create the flavor you want, AND they offer a Minimal Install disk containing a very base system you can build on.

The Minimal Install does not put ANYTHING on the disk except what is required to boot and use the base system. There are no music players, office suites, picture viewers, etc... Heck, there isn't even a desktop or graphical login. It is command-line only. HOWEVER, this can EASILY be changed. You add what you want.

I have spent a very long time compiling this information, testing it out, etc, and have decided to put it all together as a \*buntu Minimal Install Guide. Why the "\*" in front of "buntu"? Because you can turn the Minimal Install into any flavor of "buntu" you want. You can create a "homebrew", Xubuntu, Lubuntu, Kubuntu, Openbox, etc.

For this guide, I am going to show you how to do a Minimal Xubuntu Install. If you want something else, this guide will still be of use to you, and then you can alter what I have listed below in the customization section.

You can download the Mini ISO from here: <https://help.ubuntu.com/community/Installation/MinimalCD>

Get a list and description of the packages on Mini.iso (Ubuntu 15.04) here: <http://packages.ubuntu.com/vivid/ubuntu-minimal>

To see what packages are included as the Depends for

Xubuntu, you can use this command:

```
apt-cache show xubuntu-desktop | grep "Depends"
```

To see what packages are used as Recommends for Xubuntu, you can use this command:

```
apt-cache show xubuntu-desktop | grep "Recommends"
```

## CREATING THE BASE SYSTEM

Either burn the ISO to disk, or use Unetbootin to create a bootable USB thumbdrive.

Make sure your computer is connected to the Internet via an Ethernet cable.

Boot to the minimal install disk and choose "Install". Note: the "Tab" key will cycle through answers, and "Enter" will select the answer you have highlighted. In some areas, you can use the left/right arrow keys to choose between "Yes", "No", and "Go Back".

Choose your language, territory, and keyboard layout at the prompts.

Name your computer.

Choose the Mirror (simply select your country if possible).

Set proxy (or leave blank if none are needed).

This is a net install, so it will then begin downloading components to install.

Fill in the name you want to use.

Choose your username. For security purposes, you can choose a username other than your first name. Think about it, half of logging into your computer is knowing the username. The other half, of course, is knowing the password.

Choose your password. I like to make my passwords something I can remember, but are not easy to guess or emulate. For example, I



might pick a phrase like: “I hate Mondays”, but will change it so that, even if someone knows you hate Mondays, they will have a hard time figuring out how you did your password. “I hate Mondays” can turn into “lh@3m0nd&z”. This way you are using (no repeat) numbers, letters, symbols, and capitalization for your password.

You are then given a choice to encrypt your Home directory. I would say yes... this way, if anyone boots to a live CD on your system; takes your drive; etc. they will not be able to access your files in your Home Folder without finding a way to decrypt it.

Confirm or change your timezone.

**NOTE:** At this point, prior to moving on, you may remove your USB thumbdrive. The reason you may wish to do this, is because there are occasions in which Grub gets written to the thumbdrive instead of the hard drive. If you hit enter and moved on to “Partition Disks”, you can “Go Back” to the timezone, remove your USB thumbdrive, and proceed. If you continued the installation with the USB thumbdrive still connected,

and Grub was written to the USB thumbdrive... don't worry, it is an easy fix that will be listed in “Troubleshooting” at the end of this Guide.

Partition Disks: Unless you have other plans, I would choose “Guided – use entire disk and set up encrypted LVM”. This will encrypt the installation and require a passphrase in order to boot the computer.

It will then ask you to select the disk (usually there is only one choice... unless you left in your USB thumbstick). You will then need to confirm the changes to be made.

If you chose encrypted LVM, you will now need to provide a passphrase. I would do this in a similar fashion as I suggested above with the password.

You will then be asked to change or confirm the amount to use for the guided partition. I would just select continue (unless you have other plans).

You will then be asked to confirm the changes to be made.

You will now need to choose

how you want to manage upgrades to the system. I would NOT choose Landscape. Landscape is a proprietary web service. I generally choose “Install security updates automatically”, but you don't have to.

Next, you will be asked to make a software selection. Here you can choose to install a full system, server, etc. The idea here is to do an install that you fully control, so DO NOT SELECT ANYTHING. Just skip it (choose “Continue”), and move on.

You will then be asked to install the Grub Boot Loader to the MBR. Make sure it does not write to the USB thumbdrive (if you did not remove it), and choose “yes”. If it accidentally writes to the USB thumbdrive, we can fix it in the troubleshooting section.

You will then be asked to confirm the UTC clock settings.

You will then be prompted to remove any installation media and reboot.

## CUSTOMIZING THE INSTALLATION

Here is where we are going to make your system shine the way you want it. Here you can create the desktop you want with the programs you want. As stated earlier, I will be setting up a Minimal Xubuntu system. You can alter this as you get to the appropriate parts of this customization.

After rebooting, You will be brought to the command-line login prompt. Put in your credentials and log in. At this point you have a couple of options on how you wish to proceed.

You can:

- Hand-type in all of the commands.
- Run a script. (I will provide a way for you to get a script in the script section).
- Copy and paste the commands.

**NOTE:** your best bet is to use a pre-made script (discussed in the Script section).

The problem with options “2” and “3” above, is that you have no ability at this point to access a text file or script. In order to do this, you are going to have to mount a

# HOWTO - UBUNTU-BASED MINIMAL INSTALL

USB thumbdrive. The trick at this point though is... it will not auto-mount. No worries. If you want to mount a USB thumbdrive, do the following:

Plug in the USB thumbdrive.

Type in the command:

```
sudo fdisk -l
```

**NOTE:** You are looking for a partition like “/dev/sdb1”, remember what it is called (most likely “/dev/sdb1”).

Create a mount point: `sudo mkdir /media/usb`

Mount the USB thumbdrive:

```
sudo mount /dev/sdb1 /media/usb
```

Change to the USB directory:

```
cd /media/usb
```

You can now list the files contained there by typing in: `ls`

You can now run any “sh” script with the command: `sh filename.sh` (replacing “filename” with the name of your .sh file).

If you choose the copy/paste method, you may run into some issues... however, “nano” is installed by default. To edit a text file, you would use the command “nano filename” (without the quotes, replacing “filename” with the name of the file).

When you are done with the USB thumbstick, you can unmount it with the command:

```
sudo umount /media/usb
```

Here is the step-by-step commands we are going to use (refer to the Script section for automating this process):

**NOTE:** There are a couple of ways you can install from the command-line. Each one has its lovers and haters. Each one has its perks and disadvantages. They are as follows:

```
sudo apt-get install --install-recommends packagename
```

This installs a package and all of its recommends. This may install a large amount of files, some of which you may not need.

```
sudo apt-get install --no-install-recommends packagename
```

This installs the bare essentials of a package. It is a lighter install, but the program may not have the functionality you expect. You can see what requirements or recommends a package may be missing by using the command: `apt-cache show packagename`

```
sudo apt-get install packagename
```

This is the standard method of installing packages, suggested packages will be listed, but not installed.

For this guide, I am going to simply use this command (for most things): `sudo apt-get install packagename`

If you wish to do otherwise, you are free to do so.

## CUSTOMIZING THE SYSTEM (PART 1)

The packages you decide to use can be based upon your desire for security, minimal footprint (uses less computer resources), etc. I am

choosing these packages due to functionality, small footprint, customization abilities, and security.

The Beginning:

The first thing we need to do is update the Repositories with the command:

```
sudo apt-get -y --force-yes update && sudo apt-get -y --force-yes upgrade
```

The second thing we need to do is install the Depends.

**Warning (1):** I ran into a few issues during test installs. I believe I narrowed down the problems. One of the problems was that the package “ubuntu-extras-keyring” could not be found. Watch for this while running a script, as it will cause an error where the Depends will not install. I read a few posts saying that “ubuntu-extras-keyring” has been removed from the core install. It is listed in the Depends below, but it might be advisable to remove it from the install (I removed it in the Scripts section below).

**Warning (2):** Another issue I had was related to policykit-1 and

dpkg. I believe I resolved it by adding "policykit-1" to the Depends install (which was not listed in the official, original list of Depends).

It is my understanding that you can install the Depends a couple of ways, but I have not tested Method 1:

## METHOD 1: (NOT TESTED, BUT SHOULD WORK)

Type in the command:

```
sudo apt-get install --no-install-recommends xubuntu-desktop
```

This supposedly installs only the Xubuntu Desktop (you can substitute "ubuntu-desktop" or "lubuntu-desktop" for "xubuntu-desktop")

## METHOD 2: (THE ONE I TESTED AND KNOW WORKS)

First, let's install Policykit 1 to make everything else work more smoothly:

```
sudo apt-get install policykit-1
```

To install the Depends, Type the Command:

```
sudo apt-get install alsa-base alsa-utils anacron bc ca-certificates dmz-cursor-theme doc-base foomatic-db-compressed-ppds genisoimage ghostscript-x gtk2-engines-pixbuf inputattach language-selector-gnome libasound2-plugins libpam-systemd libsasl2-modules libxp6 memtest86+ openprinting-ppds pm-utils printer-driver-pnm2ppa rfkill software-properties-gtk ubuntu-drivers-common ubuntu-extras-keyring update-manager wireless-tools wpasupplicant xdg-user-dirs xdg-user-dirs-gtk xkb-data xorg zenity
```

My suggestion is that you actually use all of the Depends (except "ubuntu-extras-keyring")... but if you feel you can leave some out, so be it.

## THE RECOMMENDS

How you want to handle the Recommends is up to you, but I think there is quite a bit of room to trim the fat here (which I already have done to a degree). In the list below, I removed Recommended Packages (like "abiword", "gnumeric", etc), and moved others (like "catfish", and "blueman", etc)

to another section, to make it easier for you to have a system that is 100% bare-bones if you want it, or full of every bell and whistle you desire.

I have divided the Recommends into what I believe are your Basic Recommends and your Other Recommends. The Basic Recommends will provide the functionality you may be used to, and the Other Recommends can probably be left out without much grief (if you decide you want them, that's cool too).

## BASIC RECOMMENDS

Note: I added gdebi, aptitude, and synaptic to this list for better package support.

```
sudo apt-get install acpi-support app-install-data-partner apport-gtk apt-offline avahi-autoipd avahi-daemon desktop-file-utils gdebi gvfs-backends gvfs-fuse im-config kerneloops-daemon laptop-detect libnotify-bin libnss-mdns libpam-gnome-keyring libxfce4ui-utils policykit-desktop-privileges pcmciautils aptitude synaptic software-center update-notifier whoopsie xcursor-themes xdg-utils
```

## OTHER RECOMMENDS

```
sudo apt-get install espeak fonts-droid fonts-liberation fonts-opensymbol gcc speech-dispatcher ttf-ubuntu-font-family light-locker light-locker-settings make mugshot xubuntu-community-wallpapers xubuntu-docs xubuntu-icon-theme
```

Note: You can use "xscreensaver xscreensaver-glx extra xscreensaver-data-extra xscreensaver-screensaver-bsod" instead of "light-locker light-locker-settings" which is lightdm dependent.

## SUMMARY SO FAR

At this point, you will have a command-line OS installed on your system. There are no programs to speak of... no browser, no PDF viewer, no text program (aside from Nano), etc.

## IMPROVING THE SYSTEM

This is the point in which we will turn a very, very basic system into our workhorse. Keep in mind that you can alter most of the program options I list below. For example, instead of using xfce4-terminal for

# HOWTO - UBUNTU-BASED MINIMAL INSTALL

your terminal, you can use xterm, and so on.

Most of the items below are optional, install only what you believe you want and will use. I am simply providing enough information so that people can build a fully functional system that they are generally accustomed to, without the bloat of a lot of programs they won't use.

To Compile and Install from Source:

```
sudo apt-get install build-essential checkinstall cvs subversion git-core mercurial automake autoconf libtool pkg-config libcurl4-openssl-dev intltool libxml2-dev libgtk2.0-dev libnotify-dev libglib2.0-dev libevent-dev gcc
```

Install Archive Management:

```
sudo apt-get install unace rar unrar p7zip p7zip-full p7zip-rar sharutils udevview mpack arj cabextract file-roller unzip zip
```

Install a Terminal:

```
sudo apt-get install xfce4-terminal pastebinit
```

Install a Desktop Environment:

```
sudo apt-get install fonts-dejavu-core fonts-freefont-ttf xfce4-appfinder xfce4-notifyd xfce4-panel xfce4-session xfce4-settings xfdesktop4 xfwm4 xubuntu-artwork xubuntu-default-settings xfce4-power-manager
```

Install Desktop Environment Plug-ins:

```
sudo apt-get install xfce4-cpugraph-plugin xfce4-dict xfce4-indicator-plugin xfce4-mailwatch-plugin xfce4-netload-plugin xfce4-notes-plugin xfce4-places-plugin xfce4-quicklauncher-plugin xfce4-screenshooter xfce4-systemload-plugin xfce4-taskmanager xfce4-verve-plugin xfce4-volumed xfce4-weather-plugin xfce4-whiskermenu-plugin xfce4-xkb-plugin indicator-application indicator-messages indicator-power indicator-sound
```

Install File Management:

```
sudo apt-get install catfish gigolo thunar thunar-volman tumbler thunar-archive-plugin thunar-media-tags-plugin menulibre
```

Install a Desktop Manager (for a Graphical Logon):

```
sudo apt-get install lightdm lightdm-gtk-greeter
```

Install Print Capabilities:

```
sudo apt-get install cups cups-bsd cups-client cups-filters printer-driver-c2esp printer-driver-foo2zjs printer-driver-min12xxw printer-driver-ptouch printer-driver-pxljr printer-driver-sag-gdi printer-driver-splix simple-scan system-config-printer-gnome hplip
```

Install Bluetooth:

```
sudo apt-get install blueman bluez bluez-alsa bluez-cups
```

Install Networking Tools:

```
sudo apt-get install network-manager-gnome network-manager-pptp network-manager-pptp-gnome
```

Enhance Sound Capabilities:

```
sudo apt-get install gstreamer0.10-plugins-base-apps gstreamer0.10-pulseaudio pavucontrol
```

Install Basic Utilities:

```
sudo apt-get install gnome-system-tools gtk-theme-config guicharmap
```

Install a Browser:

```
sudo apt-get install firefox firefox-locale-en xul-ext-ubufox
```

**NOTE:** If you want to watch Netflix, you will need to install Chromium (or Google Chrome)

```
sudo apt-get install chromium-browser
```

Install Conky:

```
sudo apt-get install conky-all curl lm-sensors hddtemp
```

If you have a laptop, you may be interested in enhanced power management via TLP. You may also be interested in a great encryption program called VeraCrypt that was forked from TrueCrypt. There is also a nice password vault called KeePass2. If you are interested in these programs, you will want to add the following PPA's:

```
sudo apt-add-repository -y ppa:linrunner/tlp
```

```
sudo add-apt-repository ppa:unit193/encryption
```

```
sudo apt-add-repository ppa:jtaylor/keepass
```

You will then want to update the repositories:

```
sudo apt-get -y --force-yes
update && sudo apt-get -y --
force-yes upgrade
```

## Install Basic Applications:

**NOTE:** The following applications cover security, encryption, passwords, basic functionality, etc.

```
sudo apt-get install gnome-
calculator rsync grsync
seahorse gufw parcellite
unetbootin extlinux filezilla
veracrypt leafpad xfburn
gparted gmountiso keepass2
deluge tlp tlp-rdw
```

## Install Other Applications:

```
sudo apt-get install mousepad
libreoffice-calc libreoffice-
pdfimport libreoffice-writer
libreoffice-gtk pinta vlc
evince ristretto orage
thunderbird
```

Install Media Codecs, DVD Playback, and (if you desire) Java and Flash:

**NOTE:** A lot of this can be done via installing “ubuntu-restricted-extras”, but I wanted to give you more control.

```
sudo apt-get install
flashplugin-installer
openjdk-8-jdk ffmpeg
gstreamer0.10-plugins-bad
lame libdvdread4 libavcodec-
extra gstreamer0.10-fluendo-
mp3 gstreamer0.10-plugins-
ugly gstreamer1.0-fluendo-mp3
gstreamer1.0-libav
gstreamer1.0-plugins-bad
gstreamer1.0-plugins-ugly
libavcodec-ffmpeg-extra56
gstreamer0.10-plugins-bad-
multiverse libav-tools
chromium-codecs-ffmpeg-extra
oxideqt-codecs-extra
```

Note: as an option you can also install M\$ Core Fonts via:

```
sudo apt-get install ttf-
mscorefonts-installer
```

## Clean Up the System:

```
sudo apt-get autoclean &&
sudo apt-get clean && sudo
apt-get autoremove
```

## REBOOT THE SYSTEM

After Reboot, you can then make some final configurations and enjoy your system.

## SUMMARY SO FAR

You now have a full-fledged, working system. We can now move on to some last minute tweaks and

customization. Once you are booted up to your nice, new desktop, you will need to open the terminal and then we can finish up.

## Enable DVD Playback:

```
sudo
/usr/share/doc/libdvdread4/in
stall-css.sh
```

## Start TLP for Laptops:

```
sudo tlp start
```

Make a Directory for Adding Fonts: (Just place new fonts in this folder and they will be available to you)

```
mkdir ~/.fonts
```

Create a Directory for Source Compiling:

```
sudo chown $USER
/usr/local/src
```

```
sudo chmod u+rwX
/usr/local/src
```

## Copy Files:

**NOTE:** You will want to edit the copied .conkyrc file, not the original.

```
cp /etc/conky/conky.conf
~/.conkyrc
```

## Activate Sensors for Conky:

```
sudo sensors-detect
sudo service kmod start
sudo chmod u+s
/usr/sbin/hddtemp
```

You can get a list of all installed packages by using:

```
dpkg --get-selections >
~/Downloads/list.txt
```

## Edit Conky:

**NOTE:** There are a lot of interesting scripts you can find on the Internet. Find information at the following links:

[http://conky.sourceforge.net/config\\_settings.html](http://conky.sourceforge.net/config_settings.html)

<http://conky.sourceforge.net/variables.html>

```
leafpad ~/.conkyrc
```

That's it, we're done. Enjoy!

## TROUBLESHOOTING

Things rarely go 100% smooth and easy. If you run into a few problems that may occur during

installation, hopefully these tips can help you out.

**Problem: You accidentally installed Grub on the USB thumbdrive and now you cannot boot into Linux.**

**Solution:** Plug your USB thumbdrive back in, boot the computer, and then remove the USB thumbdrive.

Open a Terminal:

```
sudo grub-install /dev/sda
```

Reboot the computer and all should be well.

**Problem: You are booting to a black screen.**

**Solution:** Boot to Grub (hold "Shift" while booting), press "e" for Edit. Add "nomodeset" (no quotation marks) before "quiet splash" and then press F10. It should then boot to the initial command prompt. If it is still doing it after you have completed installing your system, you will need to correct drivers or permanently edit Grub.

**Problem: When you unplug your wired Ethernet cable, the**

**computer takes a long time to boot due to a "locating network" issue and/or you have no Internet (after removing the Ethernet cable) even though the wireless seems to be connected.**

**Solution:** This can be fixed by doing the following:

```
sudo leafpad  
/etc/network/interfaces
```

Comment out (#) all of the items except "auto lo" and "iface lo inet loopback"

Reboot... This should resolve the issue.

## SCRIPTS

Even though I have an IT background, I am an extreme noob when it comes to writing Linux scripts. Bash/SH Scripts are akin to M\$ Batch Files. The pound sign/hashtag (#) is used for comments.

Below I will show you how to create an Installation Script to make your life easier when installing your system. You can create a file called [whatever you want].sh and copy the text into the

file and save it. Be careful to insure that the format of your .sh file resembles my text.

**NOTE:** I would choose a name that is a single word or hyphenated text. For example: If you want to name it "Lame Script This Guy Wrote.sh", it would be better to name it "lamescriptthisguywrote.sh"... as for me, I would pick something simple like, "basicinstall.sh".

**NOTE:** There are (I'm guessing) millions of people better at writing scripts than me, and if you are one of them, PLEASE write a better one. The only issues I have had with mine are:

- Sometimes, not everything installs (probably due to me putting in an errant "return", etc). One solution I have found is to break the script down into several batches of installations instead of a huge installation segment. Breaking the scripts into many pieces also comes in handy when troubleshooting during an Install.
- Sometimes I put the cart before the horse, and some things cannot occur because the prerequisites were not met. I solved this by breaking my scripts down into multiple scripts, so that you can

launch each script after the prerequisites are done (like needing to reboot).

- I cannot automate everything (like inserting text into a specific place in a populated file), and so there are things that (at least for me) have to be done manually.

**NOTE:** If necessary, you can make the script executable by running the command: `chmod +x [filename].sh` (though I haven't run into the need so far).

**NOTE:** Run the script with the command: `sh filename.sh`

Do **NOT** run the script as sudo. The various commands will use sudo when needed.

Below is an example of how to make a Script File.

- Create a blank document, and name it whatever you want, with a .sh at the end (example: filename.sh).
- Open the empty document with a text editor (leafpad, etc) and the very first line should be: `#!/bin/bash`
- Hit "Enter" to go to the next line. From here, you can fill in the commands you want to run in the script.

## EXAMPLES OF SCRIPT FILES

If you want to change any of the programs listed, feel free to do so.

## BASIC INSTALLATION

Copy the entire segment of text from “#!/bin/bash” through to “#End of Script”.

Shown top right on this page is an example of an After Installation (and After Reboot) Script. Again,

copy the entire segment of text from “#!/bin/bash” through to “#End of Script”)

Well, that's it for now. Hopefully you will have a lot of fun creating your own Homebrew System. It is my intention to write another article expanding on this one, perhaps getting into tweaking/customizing the install, including things you can do with Firefox, writing a Conky script, etc.

```
#!/bin/bash
```

```
#Update Repositories:
```

```
echo "Updating Repositories."
sudo apt-get -y --force-yes update
sudo apt-get -y --force-yes upgrade
```

```
#Install Necessary Depends:
```

```
echo "Installing Necessary Depends."
sudo apt-get install policykit-1
sudo apt-get install alsa-base alsa-utils anacron bc ca-certificates dmz-cursor-theme doc-base foomatic-db-compressed-ppds genisoimage ghostscript-x gtk2-engines-pixbuf inputattach language-selector-gnome libasound2-plugins libpam-systemd libsasl2-modules libxp6 memtest86+ openprinting-ppds pm-utils printer-driver-pnm2ppa rfc2539 software-properties-gtk ubuntu-drivers-common update-manager wireless-tools wpasupplicant xdg-user-dirs xdg-user-dirs-gtk xkb-data xorg zenity
```

```
#Install Basic Recommends:
```

```
echo "Installing Basic Recommends."
sudo apt-get install acpi-support app-install-data-partner appport-gtk apt-offline avahi-autoipd avahi-daemon desktop-file-utils gdebi gvfs-backends gvfs-fuse im-config kerneloops-daemon laptop-detect libnotify-bin libnss-mdns libpam-gnome-keyring libxfce4ui-utils policykit-desktop-privileges pcmciautils aptitude synaptic software-center update-notifier whoopsie xcursors-themes xdg-utils build-essential checkinstall cvs subversion git-core mercurial automake autoconf libtool pkg-config libcurl4-openssl-dev intltool libxml2-dev libgtk2.0-dev libnotify-dev libglib2.0-dev libevent-dev
```

```
#End of Script
```

```
#!/bin/bash
```

```
#Activate DVD Playback:
```

```
sudo /usr/share/doc/libdvdread4/install-css.sh
```

```
#Start TLP for Laptops:
```

```
sudo tlp start
```

```
#Make Directories:
```

```
mkdir ~/.fonts
```

```
#Directory For Source Compiling:
```

```
sudo chown $USER /usr/local/src
sudo chmod u+rwX /usr/local/src
```

```
#Copy Files:
```

```
cp /etc/conky/conky.conf ~/.conkyrc
```

```
#Activate Sensors for Conky:
```

```
sudo sensors-detect
sudo service kmod start
sudo chmod u+s /usr/sbin/hddtemp
```

```
#Can get a list of all installed packages by using:
dpkg --get-selections > ~/Downloads/list.txt
```

```
#Edit Conky
```

```
leafpad ~/.conkyrc
```

```
#End of Script
```



# Get *unlimited* access to a cutting-edge technology and business library with **Apress Access!**

## For **\$199**

### YOU GET:

- Unlimited access to Apress titles for a full year
- Instant access to each new Apress publication
- Compatibility with any device—desktop, laptop, or mobile
- Use of our new exclusive-to-Apress reader with unparalleled search functions
- Option to download any eBook for just \$4.99 for a limited time



[www.apress.com](http://www.apress.com) |  @apress

Want more info? Check out [www.apress.com/subscription](http://www.apress.com/subscription)

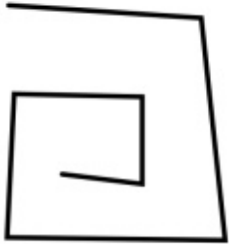




This month we're going to start looking at Live Path Effects (LPEs). These are a way to add more powerful capabilities to paths – such as drawing a pattern that follows a path (for creating ropes and chains), or rendering a path as though it's been roughly sketched. There are 13 LPEs in version 0.48, increasing to 15 in 0.91 with a slight change of UI as well.

Let's dive straight in with a relatively simple LPE: Spiro Spline.

Draw a path using Bézier tool, consisting of straight lines forming a simple spiral type of shape. Something like this:



Now select the path and open the LPE dialog using the Path > Path Effects... menu option (Path Effect Editor... in 0.48). At this

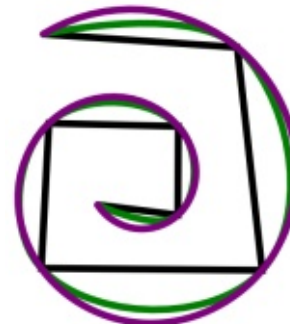
point, the user interface diverges. To add the Spiro Spline LPE to your path:

- In 0.48.x, select "Spiro spline" from the drop-down list of effects at the top of the dialog, then click on the Add button next to it.
- In 0.91, click the "+" button at the bottom left of the dialog. This will open another dialog listing the available effects. Scroll down and select "Spiro spline," then click the Add button. The second dialog will close, adding the effect to the list in the main dialog.

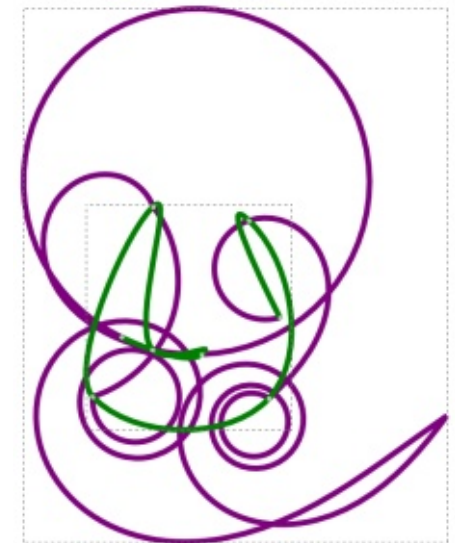
With the Spiro Spline effect added to your path, you might be a little underwhelmed to see that there's been no change to the appearance of your spiral. This is because the spiro algorithm works only on paths where some of the nodes are smooth or symmetrical. Currently, all the nodes in our shape are corner/cusp nodes. Double-click on the path to both select it and switch to the node tool, and you should see that all the nodes have the diamond-shaped handles that represent cusp nodes. Press CTRL-A to select

them all, and use the buttons on the tool control bar to change them to smooth, symmetrical or auto-smooth. Immediately you should see the effects of the spiro algorithm, as your square spiral turns into a super-smooth version.

You may be forgiven for thinking that your new spiral is just a normal result of switching to smooth nodes, but that's not the case. In the LPE dialog, you'll notice that the Spiro Spline entry has an "eye" icon next to it. Click that to toggle the effect on and off, and you'll notice that the spiro version of the path is noticeably smoother than the normal version. Here's the original (black), smooth (green), and spiro (purple) versions of the path – overlaid on top of each other so you can more easily see the differences between them.



The real difference comes when you start to manipulate the path: spiro splines are indifferent to changes in the node handles, so the most practical way to modify the path is to move the nodes themselves. The algorithm is a little unstable, and can sometimes shoot off into wild shapes as you do so; undoing your edit, or moving the nodes a little more, will generally get things back on track. For example, this image shows a green original path, plus the same path with the spiro LPE added in purple, demonstrating just how out-of-control the spiro algorithm can get!



As well as moving nodes, there's one other way to manipulate spiro paths: straighten some sections. Simply select the end nodes of a segment and use the "Make selected segments lines" button on the tool control bar to straighten it. The spiro algorithm will ensure a smooth transition between straight and curved segments. If you need to introduce a sharp transition into your path, you first have to convert one of your smooth nodes into a corner node. That alone isn't usually enough to do the job though – moving the adjacent node to one side will usually also alter the path on the opposite side in an effort to maintain the spiro path's smoothness. The secret is to move the handles of the corner node so that they're no longer co-linear, then you'll be able to move the spiro paths on either side as you would expect, with a sharp transition occurring at the corner node.

The spiro algorithm was originally created by Raph Levien for font design (see <http://www.levien.com/spiro/> for more details), but it can also be useful for flowing, organic shapes



such as plants, leaves, and... tentacles. It's so useful, in fact, that Inkscape has dedicated buttons on the Bézier and Pencil toolbars which automatically add the Spiro Spline LPE to any lines you draw with them. Select the Pencil tool and ensure that the smoothing is set to about 50% - a little either way won't make much difference. On the tool control bar (shown above), enable Spiro mode using the second button on the bar:

Now it's time to draw something on the canvas: the kind of shape that suits spiro mode. Try drawing a circle, keeping it as neat as you can, and finishing in the starting node. As you draw you'll see a green line indicating your path, regardless of your current fill and stroke settings – don't worry, that's just a guideline that won't be visible when you've finished. Unless you have supernatural control over your muscles, the green path is likely to be bumpy and distorted; yet, on releasing the mouse button, it will be replaced with a nicely rounded circle. If your

original path was extremely rough, you might not get a perfect circle, but the final shape will certainly be a lot smoother than your hand-drawn efforts.

Select the path you've drawn, and, in the Path Effects dialog, you should see that the Spiro Spline effect has been added. Toggle the visibility button (the eye-shaped icon) to see how the spiro version compares with your original path. Now try the process again with a different shape – a figure of eight, or a spiral. As you can see, for some shapes it's a lot easier to create something neat and smooth using spiro mode.

When using the Bézier tool in spiro mode, the icon on the tool control bar is the same, but the drawing process is a little different. I usually suggest drawing Bézier paths as a series of straight line segments by single-clicking to place each node, then going back in Node Edit mode to add curves afterwards. If you take that approach with spiro mode enabled, you'll get a series of corner nodes

which, as we've seen, don't really play a role in the spiro algorithm. Instead you need to click-and-drag as you place each node, in order to set the curves as you go along. Personally, I find this to be much harder to control, but give it a try to see if you're better at it than me! You can always lay down straight segments with spiro mode enabled, then explicitly change some points into cusp nodes afterwards. It's not a huge workflow improvement, but does save you a trip to the Path Effects dialog to manually add the LPE.

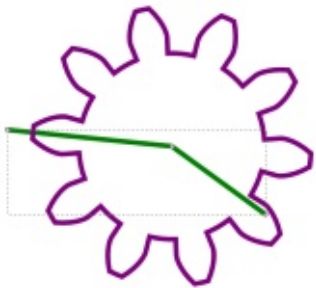
One big problem with having spiro buttons on these two tools is that it's easy for newcomers to Inkscape to enable them, then forget about it. A common question on the support forums is, "why can't I change the shape of my path using the node handles?" The answer is usually because the Spiro LPE has unintentionally been added, so watch out for that if you find yourself similarly stuck.

Let's move on to another LPE: Gears.

You first have to draw a path with at least three nodes – to begin, I'll use exactly three so that

it's clear what the relevance of each one is. With your path drawn and selected, open the Path Effects dialog and add the Gears LPE. You should immediately see your path replaced by a gear. Double-click on it so that you can see the three nodes and move them around. It should quickly become apparent that the nodes are used to set:

- 1) The angle of the first gear tooth, relative to the center point.
- 2) The center point of the gear.
- 3) The radius of the gear (from the center to the mid-point of the tooth).



With your gear selected, switch to the Bézier tool. The start and end nodes of your path should be visible: click on the end node, then double-click somewhere else on the canvas to add another segment to your path. Now the path has four nodes, and you should find that a second gear has been added, centered at the new end node.

You can repeat this process to add more nodes, and hence more gears. Applying the Gear LPE to any path with more than three nodes follows the same rules: the first three nodes define the parameters for the first gear, and any subsequent nodes set the center points for additional gears in the chain. Once you've got a few gears on screen, switch to the node tool to move their centers, noting how Inkscape automatically adjusts the radius and number of teeth in the process. Try dragging the first node around the second one to crank your gear train into life (after all, it is a LIVE path effect).

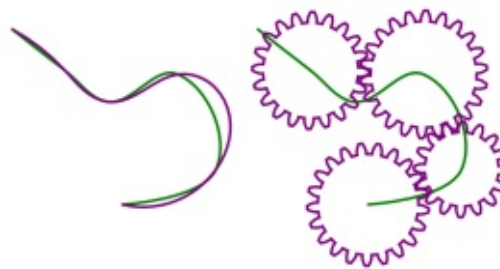
As well as the values that are implicitly set by the positions of the nodes, there are two additional parameters required to fully specify the effect. These can be found at the bottom of the Path Effects dialog, in fields labelled "Teeth" (the number of teeth on the first gear), and "Phi" (the "tooth pressure angle" - set it to about 20 for realistic looking teeth). Almost all LPEs populate this part of the dialog with a UI of some sort, and in some cases the number of additional parameters is rather excessive (watch out if

you're working on a small screen!).

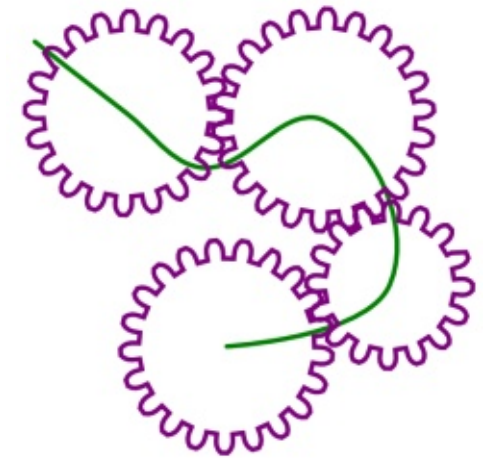
You may have guessed from the UI that it's possible to apply more than one LPE to a path. In programming terms you can think of an LPE as a function that takes a path as an input, and produces another path as an output, allowing you to chain them together. Be aware, however, that the order in which you chain them is significant. Consider this simple path, made up of cusp nodes:



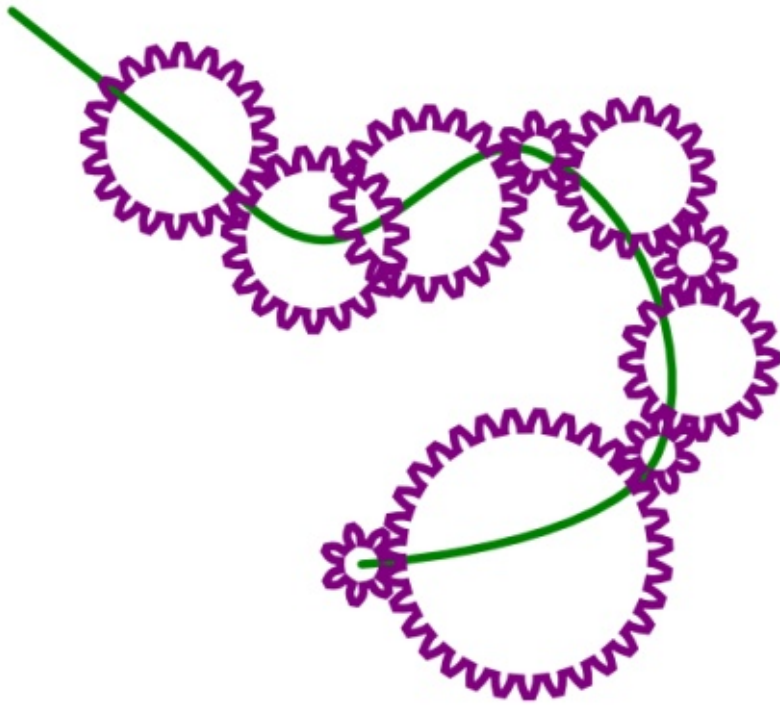
If we apply either the Spiro Spline or Gears LPE to the path we get the results we'd expect:



But if we apply both LPEs to the path, the effects differ greatly depending on the order. We'll start with the least surprising combination: Gears first and Spiro Spline second.



It may not be easy to see, but the only real effect is that the teeth of the gears have become more rounded (though each gear also has one tooth that's misshapen). Thinking about our chain of LPEs the general effect makes some sense: the first LPE outputs a composite path in the shape of the gears, then the spiro algorithm is applied to that path, smoothing out any cusp nodes that are present in it. But what happens if we apply the Spiro Spline LPE first, and the Gears LPE second?



Now we've got a load more gears! This is because the path that is created by the Spiro Spline LPE has more nodes than our original input path. Although we drew 6 nodes, the spiro version of the path actually has 13, so when the second LPE runs it creates a lot of extra gears. We could have predicted this result had we only kept an eye on Inkscape's status bar: when an LPE is active it shows the number of nodes in the output path, not the number in the original path. Try using the Gears LPE again, and have a look at how many nodes that generates!

Why not practice drawing naturalistic curves and mechanistic gears, then next time we'll continue to look at some of the other LPEs that are available.



**Mark** uses Inkscape to create three webcomics, 'The Greys', 'Monsters, Inked' and 'Elvie', which can all be found at <http://www.peppertop.com/>

## PYTHON SPECIAL EDITIONS:



<http://fullcirclemagazine.org/issue-py01/>



<http://fullcirclemagazine.org/issue-py02/>



<http://fullcirclemagazine.org/python-special-edition-issue-three/>



<http://fullcirclemagazine.org/python-special-edition-volume-four/>



<http://fullcirclemagazine.org/python-special-edition-volume-five/>



<http://fullcirclemagazine.org/python-special-edition-volume-six/>



## HAVE YOU EVER THOUGHT WHAT WOULD HAPPEN IF YOU MAKE ALL YOUR DEVELOPMENT PROCESSES RUN 10 OR 20 TIMES FASTER?

Slow builds, long running tests and scripts, compute intensive development processes delay continuous delivery, leading to longer release cycles, missed deadlines, broken builds, overworked develops, and insufficiently tested software.

## INCREDIBUILD ACCELERATES BUILDS, COMPILATIONS, TESTING, AND ANY OTHER DEVELOPMENT PROCESS

### WE SPEED UP YOUR DEVELOPMENT LIFECYCLE

Once thought a reality of every development process, **make slow builds a thing of the past.**

Increase your development productivity, accelerate your build lifecycle, and enable truly Agile development.

Realize the premise of faster Continuous Delivery and get your Continuous Integration to perform.

# ACCELERATE LINUX AND ANDROID DEVELOPMENT



*Being able to directly visually audit the build process to look for bottlenecks whilst reducing execution time is wonderful.*

**Richard Trotter**  
Geoteric



***Are you still waiting for your build to finish?***

Stop waiting. Start running. Get IncrediBuild for Linux. **Download it at [www.incredibuild.com](http://www.incredibuild.com)**



# ARDUINO

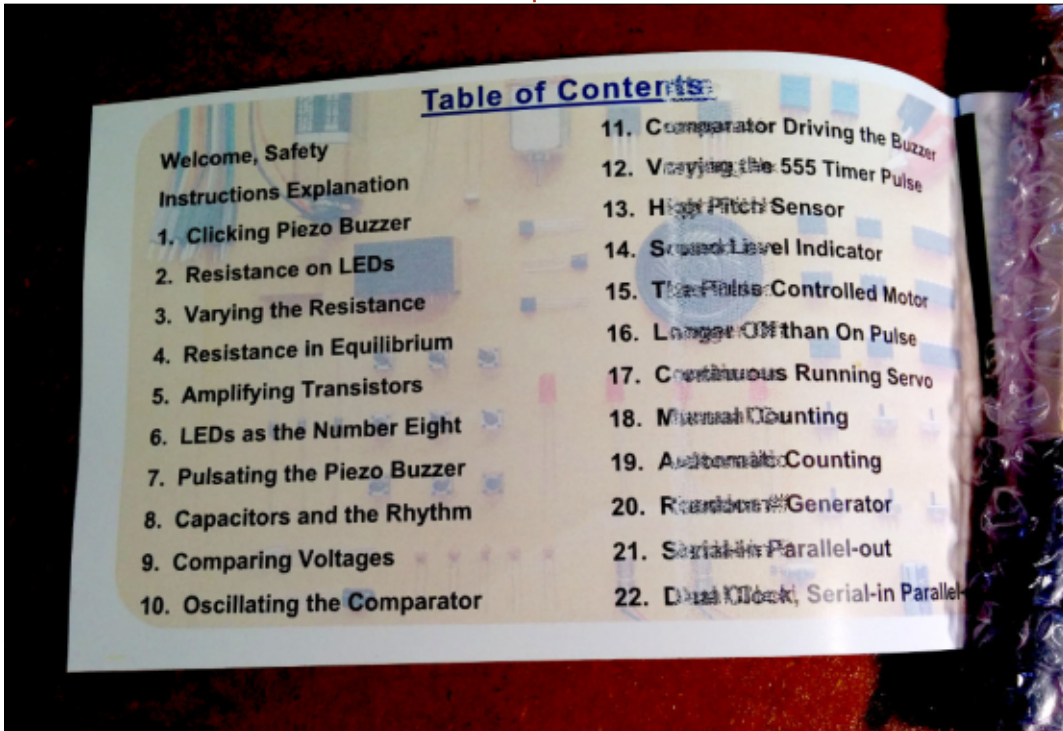
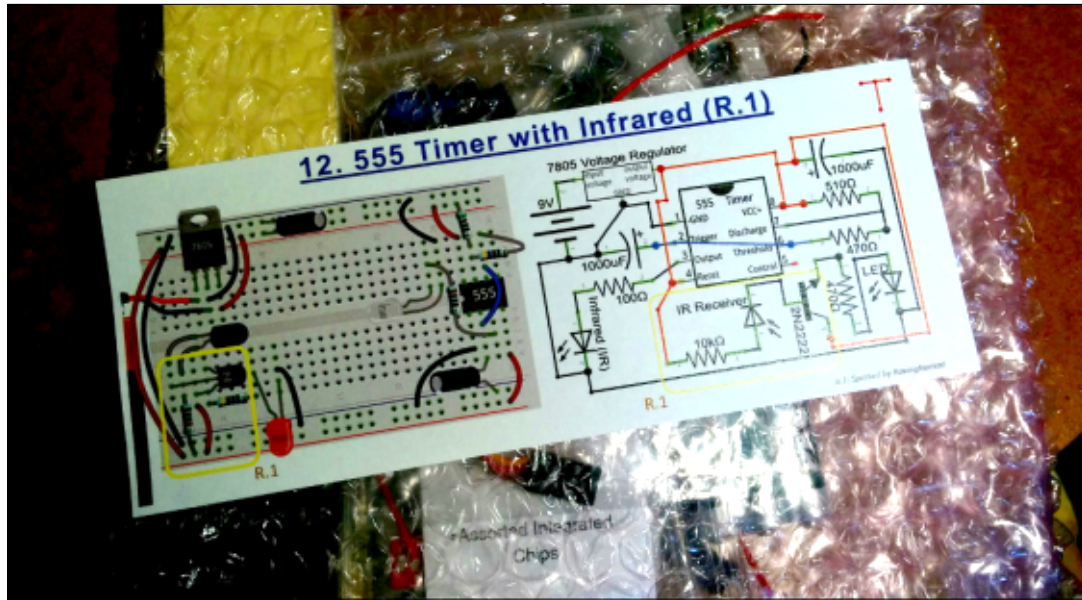
Written by Ronnie Tucker

# Tron-Club Electronics Box

Last month I showed you the first monthly subscription box from [TronClub.com](http://TronClub.com). Since I haven't done much tinkering with Arduino lately, I thought I'd show you this month's TronClub contents.

There seems to have been a bit of a print error in this month's book that I have. Some of the circuit names, on the right side of the page, are a bit smudged, but that's about all I can fault the book on.

Inside the book are several loose pages, but that's a good thing. How? Because they are corrections to this month's circuits, and also a correction for one circuit in last month's box. It's good that they're providing these as it means you can tape/glue the correction over the wrong diagram and keep the books for later reference. Thankfully, the circuit I stopped at last month is the one that's corrected this month. So now I can continue from Box 1 circuit 12.



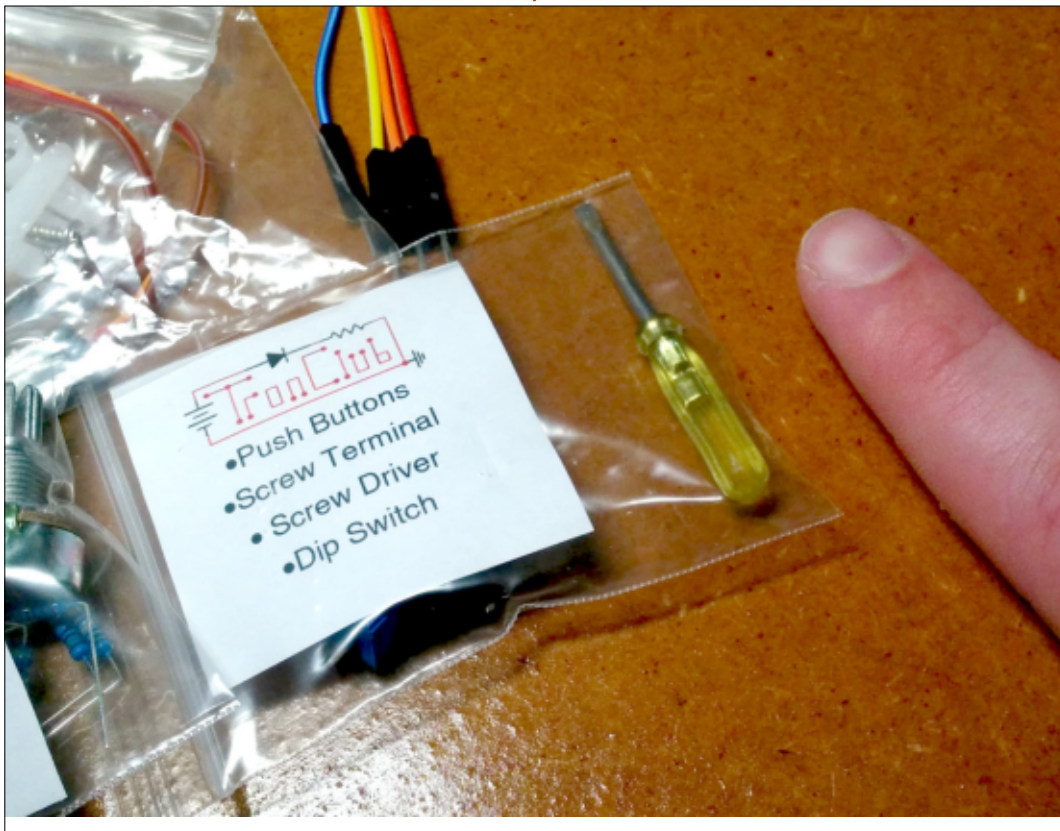
The box contents this month include the inevitable breadboard and some wires, but also a battery holder (batteries included!), some more components (IC's, buzzer, microphone, more LEDs, etc), and even a dinky little servo motor.

Speaking of dinky, my favourite part of this month's box is that it includes, quite possibly, the cutest little screwdriver ever!

### To the workbench!



**Ronnie** is the founder and (still!) editor of Full Circle. He's a part-time arts and crafts sort of guy, and now an Arduino tinkerer.

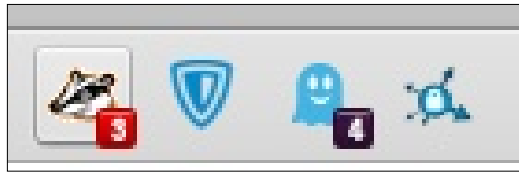






Privacy is no longer private per world governments and business corporations. However TAILS is one option to maintain online privacy, unless you have a Chromebook. The “counter-measures” the Chrome OS needs to ensure your internet privacy are: Browser Extensions, and VPN Apps. All of these items are found in the Chrome Web Store that you can install. The apps and extensions also work on the Chromium browser.

The Extensions in my Chrome browser are: Privacy Badger, Ghostery, and HTTPS Everywhere. Privacy Badger and HTTPS Everywhere were developed by Electronic Frontier Foundation. This foundation’s goal is to protect your digital rights and online anonymity. Ghostery is a proprietary freeware software developed by Evidon Incorporated. The VPN in my Chrome Browser is Zenmate. I will review these items in fuller detail. You will find the enabled extensions and VPN in the upper right-hand corner of the Chrome Browser.

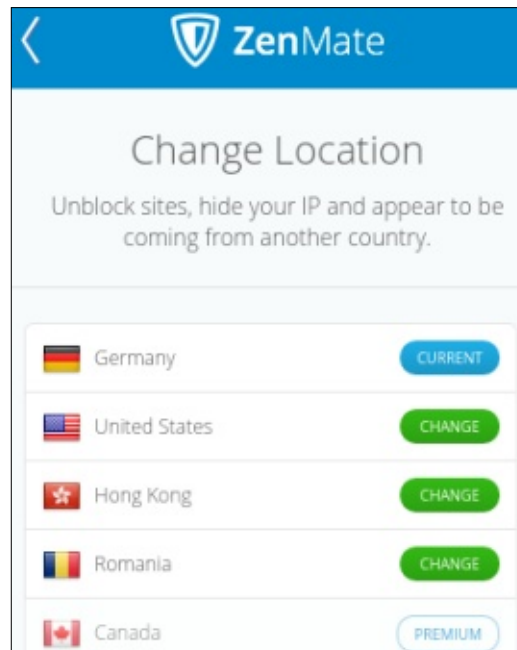


As defined by Wikipedia, a virtual private network (VPN) extends a private network across a public network, such as the Internet. It enables users to send and receive data across shared or public networks as if their computing devices were directly connected to the private network, and thus are benefiting from the functionality, security and management policies of the private network. A VPN is created by establishing a virtual point-to-point connection through the use of dedicated connections, virtual tunneling protocols, or traffic encryption.

There are many VPN providers in the Chrome Store that offer this service for free or on a monthly paid subscription. My first choice was the Tunnelbear Extension. However I quickly disliked this extension. It provided only 500MB of free service, and it has a difficult

graphic user interface. I then moved on to use Zenmate. Zenmate operates out of the United Kingdom.

Zenmate offers a free VPN app for the Chrome OS. You also have an option for a monthly subscription. Zenmate has a seamless interface. It is very easy to activate or deactivate the VPN. When the shield in the upper right corner is grey, Zenmate is off. Additionally, the free Zenmate offers various locations for the end address. Zenmate does a great job.



I would like to credit **Grant Brunner** at extremetech.com for the three extensions mentioned earlier. Privacy Badger blocks online trackers that monitor your online browsing habits. It thwarts future snooping attempts. You open the extension and can choose which trackers can be active in the Chrome Browser.



The HTTPS Everywhere extension forces websites to use the Secure Socket Layers (SSL) encryption between the web server and the Chrome browser. SSL helps keep your privacy private on a daily basis. You can connect to websites that lack SSL encryption, however any person can review your connection between the browser and the web server.






## HTTPS Everywhere

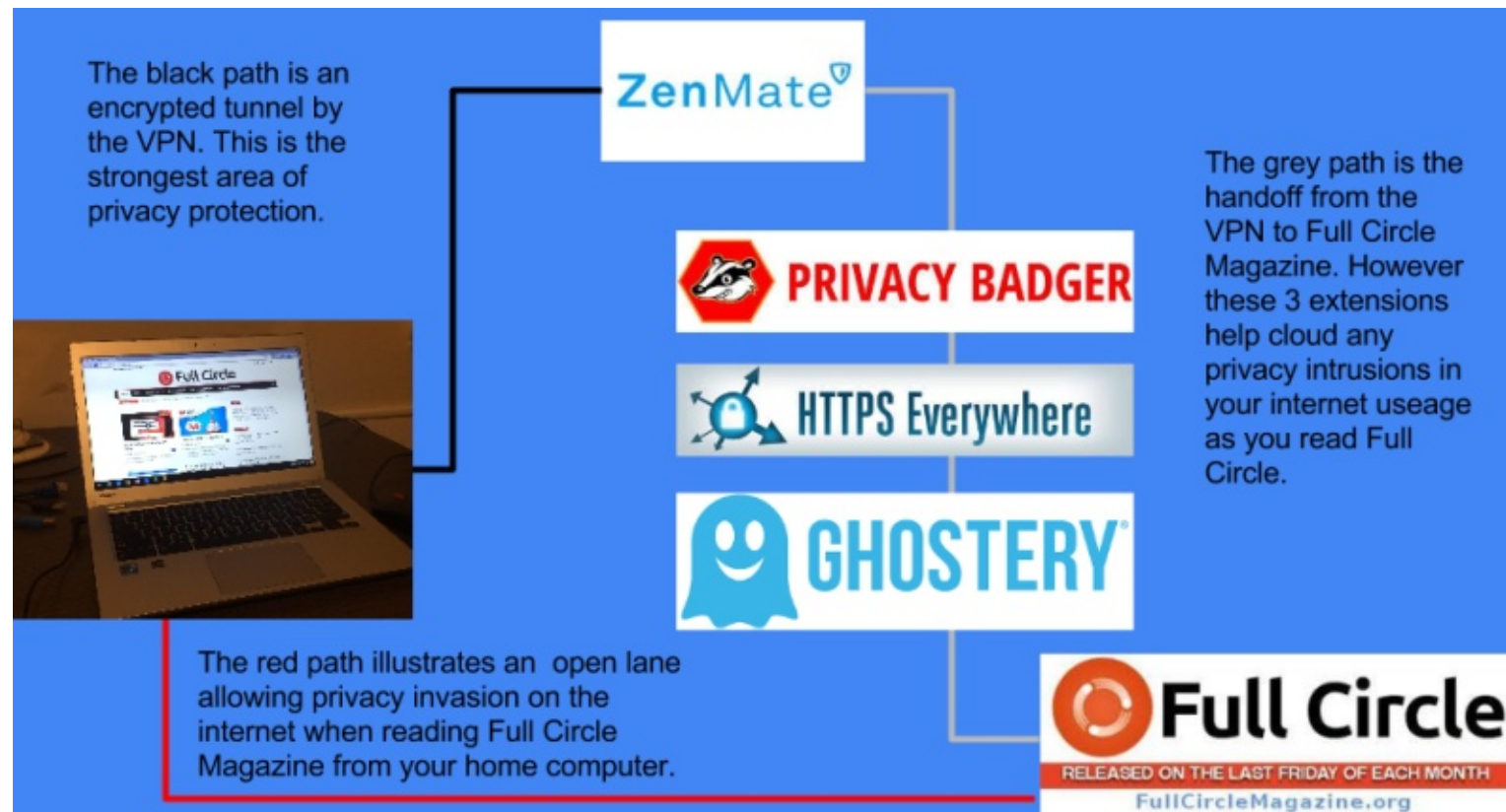
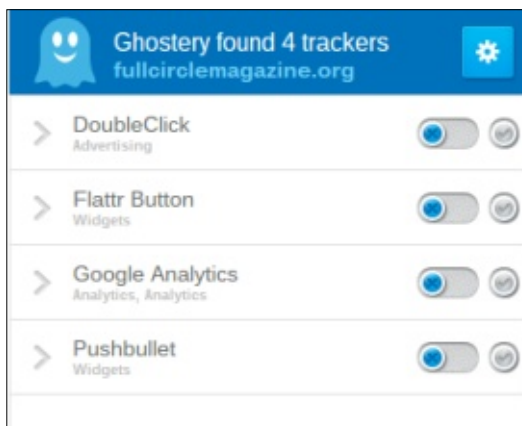
Block all HTTP requests

### Stable rules

Force [encrypted](#) connections to these websites:

-  Google APIs
-  PayPal
-  Flattr.com (partial)

Ghostery blocks HTTP requests and redirects using cookie blocking and cookie protection. It has a simple interface – similar to Privacy Badger and HTTPS Everywhere. Ghostery reports all tracking sites in the web browser within a purple box that populates in the lower right corner of the browser.



Yet Ghostery is criticized. The company will take the tracking reports and sell them to online advertisers while keeping your identity anonymous. The company calls this practice Ghostrank, and this is how it supports itself financially.

There is some overlap in the online privacy from these extensions. Yet each extension works in a different fashion. Due to the complexity of online intrusions, this varied defense is useful.

However, there are times when an extension or VPN will prevent a website from being displayed.

There are other limitations to this setup. I have yet to find all of the issues using the VPN and extensions arrangement in my browser. However I feel a bit more confident in my privacy when online. It is far from being anonymous when using TAILS; perhaps in time TOR can be brought to the Chrome OS.



Next month Chrome Cult will look into encryption tools for a Chromebook.



**SJ Webb** is a Linux Hobbyist and Research Coordinator. He enjoys fishing, hot rodding, and spending time with his kids and wife. He thanks Mike Ferarri for his mentorship.



## GUIDELINES

The single rule for an article is that **it must somehow be linked to Ubuntu or one of the many derivatives of Ubuntu (Kubuntu, Xubuntu, Lubuntu, etc).**

## RULES

• There is no word limit for articles, but be advised that long articles may be split across several issues.

• For advice, please refer to the **Official Full Circle Style Guide:** <http://url.fullcirclemagazine.org/75d471>

• Write your article in whichever software you choose, I would recommend LibreOffice, but most importantly - **PLEASE SPELL AND GRAMMAR CHECK IT!**

• In your article, please indicate where you would like a particular image to be placed by indicating the image name in a new paragraph or by embedding the image in the ODT (Open Office)

document.

• Images should be JPG, no wider than 800 pixels, and use low compression.

• Do not use tables or any type of **bold** or *italic* formatting.

If you are writing a review, please follow these guidelines :

When you are ready to submit your article please email it to: [articles@fullcirclemagazine.org](mailto:articles@fullcirclemagazine.org)

## TRANSLATIONS

If you would like to translate Full Circle into your native language please send an email to [ronnie@fullcirclemagazine.org](mailto:ronnie@fullcirclemagazine.org) and we will either put you in touch with an existing team, or give you access to the raw text to translate from. With a completed PDF, you will be able to upload your file to the main Full Circle site.

## REVIEWS

### GAMES/APPLICATIONS

When reviewing games/applications please state clearly:

- title of the game
- who makes the game
- is it free, or a paid download?
- where to get it from (give download/homepage URL)
- is it Linux native, or did you use Wine?
- your marks out of five
- a summary with positive and negative points

### HARDWARE

When reviewing hardware please state clearly:

- make and model of the hardware
- what category would you put this hardware into?
- any glitches that you may have had while using the hardware?
- easy to get the hardware working in Linux?
- did you have to use Windows drivers?
- marks out of five
- a summary with positive and negative points

**You don't need to be an expert to write an article - write about the games, applications and hardware that you use every day.**



# The Fourteenth Annual Southern California Linux Expo

<http://www.socallinuxexpo.org>  
Use Promo Code FULL for a 30%  
discount on admission to SCALE

# SCALE 14x

The Southern California Linux Expo has grown in size and scope since it began, and given this trend we will be in a new venue as of 2016.

We're happy to announce the dates and location for SCALE 14x...

January 21-24, 2016

Pasadena Convention Center

Pasadena, CA

Featured Speakers:

Jono Bacon

Jon "maddog" Hall

Cory Doctorow

Bryan Lunduke





**B**uilding a dedicated computer for wiping hard drives is not as complicated or expensive as you might imagine, thanks to the free software Darik's Boot And Nuke. With a minimal amount of hardware, you can build a machine dedicated to wiping both SATA and PATA hard drives.

This article is intended for people who have a need to wipe a lot of hard drives, or just have a spare system lying around they don't know what to do with. If you need audit-ready reporting for regulatory compliance, or SSD support, then you might want to check out Blancco software (they also make the free DBAN software).

You might wonder "why build a dedicated machine when you can just pop a DVD into each machine and wipe the machine that way?" Several reasons:

- Donated machines don't always function, so you might have to pull the hard drive and erase it outside of the donated machine.
- Having 6 machines wiping hard

drives takes up a lot more energy than using 1 machine to wipe 6 drives.

- You develop a good workflow of removing and testing drives.
- It gives volunteers (if you're a refurbishing project) another task they can do without needing to know all the details of building a machine.

## MATERIALS YOU'LL NEED

The materials you'll need will vary depending on what you have available, whether you're doing this solo or have volunteers helping you, and how multi-purpose you want to make your machine. This list is by no means exhaustive, but I'm including a bit

more than you need to start with:

- a motherboard with SATA and PATA headers (you can use one with just SATA or PATA but having both onboard simplifies things).
- PCI/PCIe SATA/PATA expansion card (if you want to add more SATA/PATA headers, you can get cards with more connectors than the card linked to here, this was a cheap card) (optional).
- a good power supply unit (500W or better recommended).
- a SCSI controller card if you want to wipe older 50/68 pin SCSI drives (optional).
- DBAN (Darik's Boot and Nuke).
- SATA data cables.
- PATA data cables (finding one in a store might be tough but you can find lots of old machines with

them).

- Molex to SATA Y power cable (optional, for expansion if you have an older power supply).
- DVD-ROM drive (to boot DBAN from).
- Docking module for IDE (optional).

You can put DBAN on a multi-boot USB key, but since USB keys tend to be writeable, you risk overwriting your USB key when you run DBAN. Using a CD/DVD to boot DBAN eliminates the potential of overwriting your media. If you want to get really fancy, you can set up a server and PXE boot (network boot) DBAN, but this is beyond the scope of this article. The idea here is to get you started as fast as possible.

## STEPS

- Build your drive wiping machine (hardware side).
- Burn the DBAN ISO to a CD/DVD.
- Set your DBAN machine BIOS to boot from CD/DVD first.
- Connect your drives and run DBAN.



## BUILD YOUR DRIVE-WIPING MACHINE (HARDWARE SIDE)

We started our build with a MSI 945GZM3 (MS-7267) motherboard. This motherboard was one of many motherboards sitting around the shop which we had fixed capacitors for. We picked this motherboard for a few reasons:

- It supported a dual core processor.
- It used DDR2 RAM.
- It had 4 SATA headers on the motherboard and 1 PATA header (which we didn't use).
- It already had a dual-core processor and heatsink+fan installed.
- It had easy to read headers on the front panel.

We had motherboards that supported more PCI slots which are handy if you want to use a lot of PCI controller cards for more IDE/SATA ports, but in our experience we usually don't DBAN more than 6 hard drives at once. (Both because of power, and because, if one drive is bad, the rest slow down too). We had a couple of 1GB DDR2 RAM sticks around which we put into the 2

RAM slots. The case we chose was an empty non-branded mid-tower ATX silver case. We used zip ties to hide the front panel sound and USB connectors behind the ridge of the case because we don't use the front ports in our DBAN machine.

Our power supply's motherboard molex connectors were so short that we had to use a zip tie to keep it from falling into the CPU fan. Two molex Y to SATA power connectors were used to provide extra SATA power headers. Then we added all the SATA cables and a SATA controller card plus two more SATA cables for a total of 6 SATA data ports. If you have more controller cards, you can add more cables but remember you're going to have to power all those hard drives! According to Superuser.com, each hard drive uses approximately 25 watts.

Because we've been around awhile, we have lots of other controller cards and useful adapters. We added a PCI IDE controller card for an extra 2 cables (4 IDE drives). As a rule, we don't DBAN PATA and SATA together; doing so tends to create issues.

On the first dban machine we ever built, we used docking modules, but we found that over time, even with training, the modules would get misplaced or ruined. Some docking modules had to be locked for a drive to be recognized (we got around this by soldering the two wires leading to the locking mechanism together so they were always locked), but perhaps the most annoying problem was that it just took too much time to put the drives in the docks. If a PATA drive wasn't jumpered correctly, we'd have to pull it out of the dock and reinsert it. Drives hanging out the side of the machine aren't pretty, but it's simple for volunteers to connect and disconnect drives.

## BURN THE DBAN ISO TO A CD/DVD

When you download DBAN, you get DBAN in an ISO format. You won't be able to just copy the file to a DVD, you need special software like Nero (Windows), K3B (Linux), or Brasero (Linux) to burn the ISO to CD/DVD. Nero, K3B and Brasero know how to handle ISO files so they get properly unpacked

to the CD/DVD. DBAN is small so it can easily fit on a CD.

## SET YOUR DBAN MACHINE BIOS TO BOOT FROM CD/DVD FIRST

Setting your machine to boot from CD/DVD first might seem like a simple task, and if you're used to a particular machine, it is. But there are a lot of motherboards out there and manufacturers often do things differently from one another. Just getting into the BIOS can be tricky, especially if the computer is fast and the manufacturer has chosen to display a splash screen instead of the hotkeys for booting to another device or entering the BIOS. In general:

- Dell tends to use F2, Del, or Enter,
- IBM tends to use F1 or Enter,
- HP/Compaq tend to use F10,
- Just about everyone else uses the Del key.

Hitting the right key before the operating system loads is... key.

Once you're in the BIOS, most systems just let you change the boot order to make CD/DVD the

first device. A few BIOSes also require that you set another setting in another spot (which can vary) to enable booting from devices other than the hard drive. If you've set your system to boot from CD/DVD first, and it isn't booting, check first to see the BIOS recognizes the drive, then look through some of your other BIOS menus to make sure there isn't another option you need to set to boot from CD/DVD. These special cases are most often on business-class systems where manufacturers recognize that system administrators don't want just anyone rebooting the machines with a CD/DVD/USB key in them.

If you can boot to your DBAN CD/DVD, you're set. If not, check the DVD. If you see only the ISO file on the DVD, it hasn't been burned correctly; re-burn with K3b or Brasero. The DVD should contain many files.

## CONNECT YOUR DRIVES AND RUN DBAN

Serial ATA drives are straightforward, 1 SATA hard drive per cable. PATA or IDE drives are a

bit more complicated because you can have more than one drive on a cable and the drives need to be "jumpered" correctly. With 2 drives on a cable you have 2 options: Master/Slave or both drives set to Cable Select. We found the simplest method that worked when training new volunteers was just to instruct them to set all hard drives to cable select, and let the cable determine which was master and slave. Again, for SATA drives this isn't an issue.

Darik's Boot And Nuke has several options for wiping. If you simply want to wipe all the drives attached using a standard 3-pass solution, type: autonuke. The F3 key displays other methods of wiping including dod (Department of Defence 5220.22-M), dodshort (the default method, 3 passes), ops2 (RCMP TSSIT OPS-II method, 8 passes), gutmann (35 passes), prng (PRNG stream), or a quick (1 pass).

In our region of Ontario, Canada, our refurbishing certification body, the Ontario Electronic Stewardship, mandates that drives we wipe for reuse be wiped with at least the dodshort (3-pass DoD 5220.22-M) method. Some donors may request a stronger method. At least a couple of donors have asked us to use the ops2 (8-pass method) on donated drives.

If you're an individual or small organization repairing computers, you may want to consider using a quick method if you're just wiping malware in addition to the OS off a drive. One pass is much shorter than three.

The amount of time dban takes to wipe a drive depends on the method chosen, the size of the hard drive, and if the drive contains any bad sectors or other errors. Drives with bad sectors can take a lot longer to wipe. A 1TB hard

drive took us several days to wipe using the ops2 (RCMP 8-pass wipe). A 3-pass wipe on the 1TB took us a full 8-hour shift. If you're dealing with a lot of large drives you may want to check to make sure they don't have bad sectors first.

To determine whether a drive has bad sectors or not, you can use a manufacturer's tool like Seagate's SeaTools, or an open source solution like Gsmartcontrol. We prefer using open source tools – both for licensing reasons and because they tend to be simple to set up on our PXE boot server. Any Ubuntu DVD/USB key can be used to test drives with gsmartcontrol, but you'll have to install gsmartcontrol in the live environment:

```
sudo apt-get install gsmartcontrol
```

When gsmartcontrol loads, all drives attached will be displayed (including DVD drives). To see the smart information about any drive, double-click on the hard drive. A new window opens with 6 tabs: Identity, Attributes, Capabilities, Error Log, Self-test Logs, and Perform Tests. Click the Perform

### Darik's Boot and Nuke: Quick Commands

You may enter these commands at the boot prompt. In each the computer will be wiped automatically without confirm

```
dod      Wipe all disks with the DoD 5220.22-M method
dodshort Wipe all disks with the short DoD 5220.22-M
ops2     Wipe all disks with the RCMP TSSIT OPS-II
```

Tests tab to run a test on a drive. You can perform 3 different tests: a Short Self-test (one-minute to two-minute test) designed to show most errors without running a complete surface scan, an Extended Self-Test (86 minutes+) which runs a complete surface scan and runs different routines built-in to the drive, and a Conveyance Self-test (approximately 2 minutes) designed to indicate if there was any damage during transportation of a hard drive.

The short test isn't comprehensive, but it's usually the best test to run to determine if the drive has any serious errors. All of the tests write to the Self-test Logs tab once the test is complete. Any errors show in the Error Log and Attributes tab. If an error appears it's important to read the complete text of the error on the Attributes tab. If you hover over an attribute in pink/red, a text pop-up appears explaining the error. It might take some sleuthing to determine how serious the error is. Generally, any errors in red are serious failures. Pink attributes: you'll probably want to get more information about these to determine whether they're serious or not. Some attributes are quite

handy to look at when building systems (Airflow Temperature for example).

Our project's process is to run the short test. If a drive fails the short test, it's physically destroyed. If it passes the short test but displays errors, we examine the errors to determine if the errors are non-serious (e.g. the computer was shut down improperly and didn't completely write to the drive) or serious. Depending on the size of the hard drive, we might perform an extended test (on a 500GB-1TB we might run a longer test if we're not sure the drive has a more serious issue).

When wiping hard drives, it's normally a good idea to try to wipe drives that are the same size to keep the wiping time down. An 80GB hard drive will wipe much

faster than a 500GB hard drive. Successfully wiped drives show SUCCESS both on the wiping screen (while a larger/slower drive is still wiping), and on the completed screen (when all drives finish). Drive model and serial numbers are displayed on both screens, so, if a drive fails, it's easy to determine which drive has failed provided you can read the serial number and model on the drive's physical label. In the screenshot, the first drive is a Seagate (we know from the ST380815AS model number) hard drive with a serial number of 6RA2G57W. For particular donors, I normally create a spreadsheet with the drive model, size, serial number, and method used to wipe the drive, along with our project's information and my name and signature to state that I've been present to see the drive's wiped.

Building a dedicated wiping machine can be as simple as using an existing machine and booting from a DBAN CD, or as complicated as a system with several expansion cards (IDE, SATA, SCSI), molex power splitters, and docking modules. We used what was on hand and we found that simplicity is often best, especially because we have a lot of different volunteers and have a lot of drives to wipe. Darik's Boot and Nuke can wipe drives using a number of different methods, but the default 3-pass DoD method is thorough enough that it satisfies some waste/refurbishing governing bodies (of course you should always check for your area if you're professionally refurbishing computers). We've used tools like foremost (created by the NSA) and Recuva (a Windows tool from Piriform, the same company that makes the popular CCleaner tool) to check wiped drives, and neither have been successful recovering any data.

Dban - <http://www.dban.org/>

```
Options
Entropy: Linux Kernel (urandom)
PRNG: Mersenne Twister (mt19937ar-cok)
Method: DoD Short
Verify: Last Pass
Rounds: 1

Statistics
Runtime: 00:00:28
Remaining: 05:19:07
Load Averages: 2.97 0.74 0.25
Throughput: 205101 KB/s
Errors: 0

ATA Disk ST380815AS 3.AA 74GB 6RA2G57W
[00.43%, round 1 of 1, pass 1 of 3] [writing] [65431 KB/s]
```





# UBUNTU PHONES

Written by Ronnie Tucker

## OTA-7

The full list of Ubuntu Phone updates are provided below:

## SCOPES

- Improved social media handling – support for ‘Likes’ and ‘Retweets’

## BROWSER

- Add search to history view
- Improved context menu with options to download links
- Http basic auth support

## GALLERY

- Support SVG format
- Soundcloud webapp now plays in the background

## BUG FIXES

- Fix for test.mmrow exploit <https://launchpad.net/canonical-devices-system-image/+milestone/ww40-2015>
- Fixes for the UI freezing (FD leaks)

- Does not create crash reports on stable channel by default
- Fix the QML cache and restore consistent app startup times
- Fixes to use less memory by default in the browser and avoid webapps showing a white screen
- Improvements to screen banking, use of proximity sensor

## UPDATE ON UBUNTU PHONE SECURITY ISSUE

A security vulnerability has been discovered on the Ubuntu Phone. We take security very seriously, and want to provide clear information as to what happened; and what steps have been taken to rectify the issue and protect against future similar incidents.

At this point, we believe that the core issue has been addressed. An app which exploited the issue has been removed; the 15 people who installed that app have been contacted; and a fix for all Ubuntu Phone users will be released

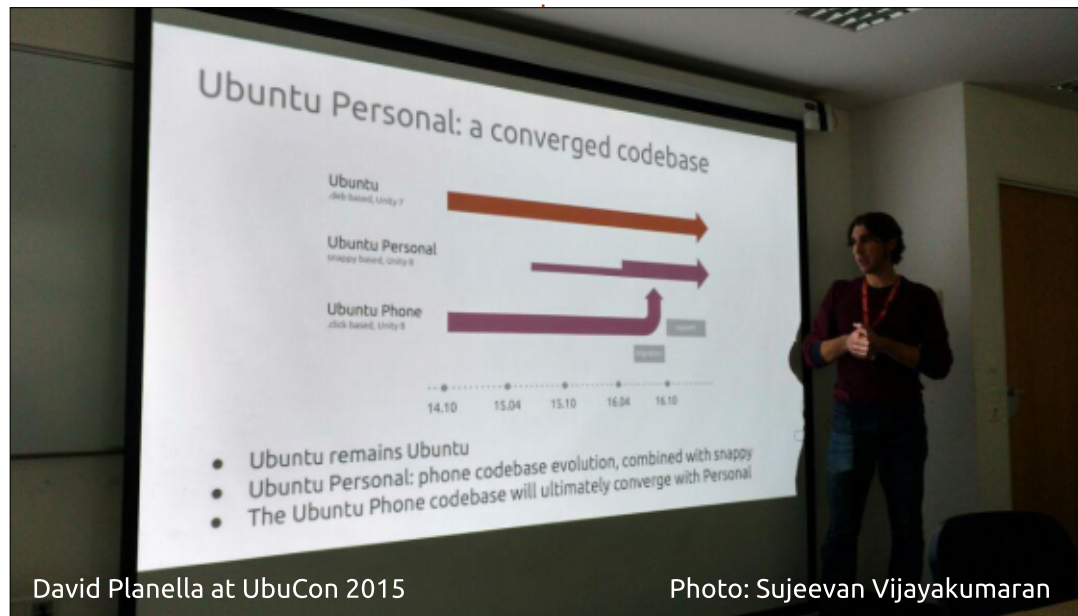
shortly. Users of Ubuntu on the desktop, server, cloud and snappy Ubuntu Core devices are not affected.

Full story at: <https://insights.ubuntu.com/2015/10/15/update-on-ubuntu-phone-security-issue/>

## UBUNTU, UBUNTU PERSONAL AND UBUNTU PHONE

UbuCon 2015 revealed that while Ubuntu would remain as

Ubuntu as we know it (with .deb files and Unity 7) the current Ubuntu Phone OS will merge with what is now known as Ubuntu Personal (with Snappy and Unity 8). This means that Ubuntu Personal will be the convergence OS that will run on both desktops and phones whereby a phone can be plugged into a display device and be used as a desktop machine would be.



David Planella at UbuCon 2015

Photo: Sujeevan Vijayakumaran

# DON'T BE A CASUALTY THIS BLACK FRIDAY

This year avoid the high street chaos and shop from the safety of your home.

Put the kettle on, relax and visit [ebuyer.com](http://ebuyer.com) for the best deals on laptops, TVs and electricals. Is Black Friday really worth a black eye?

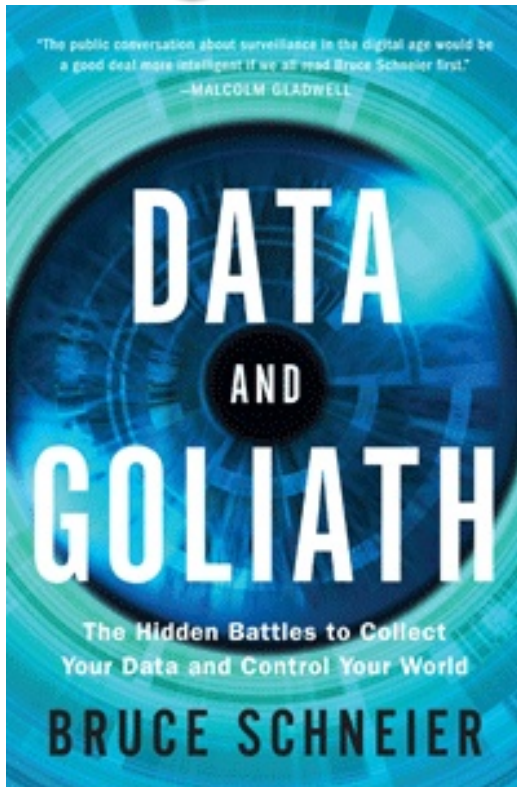
**ebuyer.com** SHOP SAFE

FIND OUR EARLIEST DEALS ON **ORANGE THURSDAY**

ORANGE THURSDAY

**26**

NOVEMBER



## Data and Goliath

by Bruce Schneier

W. W. Norton & Company

320 Pages

Hardcover

ISBN: 978-0393244816

If you have any interest in computer/data security, you probably already know the name Bruce Schneier. You may have

visited his blog, Schneier on Security, or read one of his previous books – which number in double digits – attesting to both his knowledge and longevity in the field. You can find him in many YouTube videos such as NSA Surveillance and What To Do About It - Bruce Schneier. Or maybe take a look at The Schneier Model (Kevin O'Brien, Full Circle Magazine #101, p48). His most recent written offering is Data and Goliath, and will be of interest to those with a need to know, but will likely interest many more – given the growing fear of our Internet-connected world.

With an almost daily calamity, exploit or cause for concern, the entire planet knows there is a problem with big data even if we can't articulate its nature. Big doesn't begin to describe how extensive and overwhelming it is, and, more importantly, what it will enable in the future. When machines can predict your actions and reactions better than you can, at what point do we lose control?

Nevertheless, Data and Goliath is here to save the day, albeit with some strings attached. His approach is divided into three sections:

- The world we're creating
- What's at stake
- What to do about it.

There is no doubt about the immense potential good that this represents, but he asks about the costs and security consequences. Of course plenty of facts are cited but the importance of this book is that he questions everything with the careful eye of someone who knows the field and has the experience to recognize where real balance is needed. For example, he notes that the U.S. National Security Agency does in fact purchase zero-day exploits, and the whole world would be better off if they simply released them to the computer industry for patching. But, he recognizes that a security offense capability may at times be the only viable option, and should exist along with a palpable defense. Thus he suggests the NSA release most

exploits and keep a select few for when they are demonstrably needed.

While the U.S. has the capacity to arbitrarily save all data, it makes as much sense as the ill-conceived military philosophy expressed as: Kill them all, let God decide. First and foremost, it turns a democracy into a surveilled society which inhibits progress and suppresses conversations considering change. Unfortunately, this tactic is used with purposeful effect on populations around the globe. Schneier explains all facets of this issue including impacts on human rights and liberty. And it's costly. At \$72 Billion a year for the U.S., it impacts both domestically and internationally. If you can't trust a country's security policies and laws, why would you think you can trust software or data security therefrom? Thus the NSA in the U.S. has been likened to "an autoimmune disease, because it attacks all other systems." And most obvious, if apparently unappreciated, is that the more we save, the more difficult it is to keep

it all secure, a problem for which we need no reminders.

As noted above, this book can save the day, but only if we are willing to do something about it. Politicians are unwilling to control excess surveillance because without push-back from the electorate, they respond like David Cameron. He said "I am simply not prepared to be a prime minister who has to address the people after a terrorist incident and explain that I could have done more to prevent it." Schneier makes a critical comparison with organized crime, saying "Terrorists don't cause more damage or kill more people; we just fear them more."

Likewise, corporations should be more accountable and not let "Externalities limit the incentive for companies to improve their security." Without fiscal responsibility, the only ones hurt are those providing the data – who are usually paying for the privilege in one fashion or another. Poor decisions by the very large automotive, air transportation or food processing industries to name a few are nevertheless liable, so why should big data and related

industries be any different?

Finally, the individual also has to play a larger role until much better security is baked into the industry as a whole. A variety of specific options are noted and worth considering/using by individuals wanting to have an impact. The book is much more than an insightful comprehensive look at the problem, it's also a call to virtual arms with Schneier identifying what Government, Corporations and the Rest of us need to do. Near-term, it can only get worse, but if it doesn't get better, we have only ourselves to blame.



**Jon** is an advocate and user of open source software. Known by some as the doctor of chocolate (PhD in Food Science), he is employed doing computer support at a university in the SE U.S. He is still impressed by how easy Linux distros are to install.



# MY STORY

Written by Jaideep Tibrewala

**M**y journey started with Unix when I was a student at UW-Madison. As a Computer Science student in the 90's, all CS course work was done on Sun Sparc/Solaris machines. I was pretty fascinated by this Unix world, and the different flavors available at college, which included HP-UX and DEC. They just felt more powerful in computing power than Windows at that time. However, it was not practical to buy a UNIX machine for personal use, nor was any of my class homework possible on Unix.

The UW-Madison CS department had an underground lab where the geeky CS students spent hours doing research work. Many of the lab machines ran an alternative operating system called Linux. This was the alternative to UNIX that I could bring home. So, sometime in 1997, I decided to install Redhat on my desktop PC. Installation went smoothly and my experience with Linux started.

My early years with Linux were

not very smooth. I wasted many hours trying to get a custom kernel to compile. Then get modem drivers, the sound card, graphics card, Quake 3, and so on to work. Thanks to the various Linux forums and volunteers for their guidance. Linux was still very immature at that point in time, but I wasn't giving up. Small victories gave me a sense of achievement. When Redhat spun off and created Fedora, that was my first change. I stuck with Fedora for quite a few years, and had gotten comfortable using KDE and some of the cool features it provided (eg: Amarok).

After a while, I got tired of the plug-and-pray world of Fedora, with things breaking from one upgrade to another, and decided to research other Linux flavors, with a focus on something that is a lot more user-friendly and with hardware and peripherals working out-of-the-box. That's where I came across Ubuntu. My first installation was Kubuntu Feisty in 2007, and I instantly preferred it over Fedora. The environment was cleaner and worked a lot more

seamlessly with hardware.

Over time I jumped from Kubuntu to Ubuntu and finally to Xubuntu. I realized that I needed something that was lightweight on RAM, and worked efficiently on old hardware/laptops. I stopped caring about bells and whistles from KDE or Gnome. I don't do programming any more, but am comfortable working with the command-line when required.

I currently dual-boot my laptop and live in Windows during the week, and switch to Xubuntu during the weekends. Now that almost everything works as well in Ubuntu, I prefer to stick to Ubuntu. For the last three upgrades, things have gone very smoothly and not disrupted my dual-boot environment.

What I like about Xubuntu is that it's fast, has a good interface, allows me to mount my NTFS data partition that I share with Windows, and doesn't slow down the system with unnecessary background services. Most of the

apps that I use 90% of the time on my laptop work very well in Xubuntu. Gimp is a great replacement for beginner photo editors like me. And I have the best set of rotating wallpapers thanks to Variety and wallhaven.

What I don't like about Ubuntu - there are still some things which I miss from Windows. I haven't been able to find a good linux app that will do a BPM analysis of my songs AND store the value in the respective mp3 files, nor a good replacement for a WYSIWYG app like Dreamweaver. I'm a big Excel geek, and LibreOffice or OpenOffice just don't compare, so I have to subscribe to Crossover Linux to install and use MS Office. Java doesn't work in Chromium. I can't get Quicken to work in Xubuntu or Crossover. And, in some ways, the Windows UI is a lot crisper than my current setup.

Nonetheless, I enjoy using Xubuntu and hope to continue being a devoted user for a long time. Congrats to FCM, and I look forward to reading 100 more.





Join us on:



[goo.gl/FRTMI](https://goo.gl/FRTMI)



[facebook.com/fullcirclemagazine](https://facebook.com/fullcirclemagazine)



[twitter.com/#!/fullcirclemag](https://twitter.com/#!/fullcirclemag)



[linkedin.com/company/full-circle-magazine](https://linkedin.com/company/full-circle-magazine)



[ubuntuforums.org/forumdisplay.php?f=270](https://ubuntuforums.org/forumdisplay.php?f=270)

## CHECKING FINANCES

Could you please do an article about a finance program that would be suitable for the average person? Something not just for accountants.

**Gordon Loughnan**

Ronnie says: *Anyone out there with knowledge of accounts software and wants to write something up? Email it to: [articles@fullcirclemagazine.org](mailto:articles@fullcirclemagazine.org)*

## BLUETOOTH WOES

Back in the day (prior to Ubuntu 14.04 I think), I never had a problem connecting a Bluetooth device to my laptop. Since then, no luck at all. I am a dedicated Xubuntu user and I really would like Bluetooth to work.

Downloaded the 15.10 beta, tried it hoping this problem had been fixed. No dice. The devices pair OK but they cannot connect.

Why, with an organization seemingly as adept as Canonical, has there been no progress on the Bluetooth front.

Temporary or permanent fixes abound but that begs the question rather than answering it.

I've tried numerous fixes gleaned from multiple blogs and forums – with no luck. Can you shed any light on this issue?

**Dick Smith**

## PYTHON JUBILEE

Thank you very much for the LibreOffice Golden Jubilee Edition. Everything in the same place. Very convenient! Would it be possible to do the same with the Python series, please? A Python Diamond Jubilee Edition would be great!

**Sylvain Pelletier**

Ronnie says: *Brian says he'll try to make a Python jubilee edition.*

## EPUB#101 101

A couple of people noticed two minor problems with the FCM#101 EPUB:

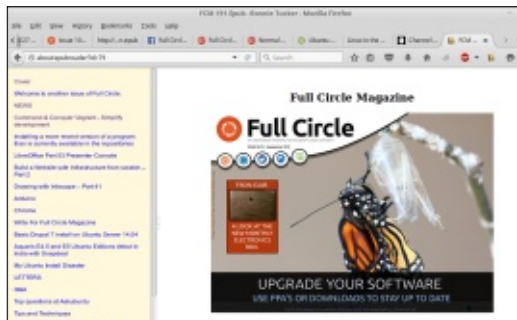
- The meta data shows it as issue 191 not 101
- The cover image is not marked as 'cover' and so does not show when imported into Google books

Brian says:

Point 1) Guilty as charged.

In mitigation, I can only say that I entered the Metadata by candle light on my laptop and the 9 and 0 are adjacent to each other.

Point 2) piqued my interest. A download from the FCM site looked like this.



## FULL CIRCLE NEEDS YOU!



Without reader input **Full Circle** would be an empty PDF file (which I don't think many people would find particularly interesting). We are always looking for articles, reviews, anything! Even small things like letters and desktop screens help fill the magazine.

See the article **Writing for Full Circle** in this issue to read our basic guidelines.

Have a look at the last page of any issue to get the details of where to send your contributions.

And the code for the cover page is:

```
<body>
  <h1 title="Cover">Full
  Circle Magazine</h1>
  <p></p>
  <h3
  class="sigil_not_in_toc">Issue
  #101</h3>
</body>
```

Everything looks OK so far.

As I know nothing about Google Books, I decided to open it. Of course the app contains no information on how to get an ePub into it. The device I was using then decided to commit suicide by hurling itself onto a concrete floor.

A search revealed: "As of March 2013, Google Play Books supports third-party ePub or PDF files. You can upload books to your account by visiting <https://play.google.com/books/uploads> in your web browser (when logged in to your Google account, of course). You are permitted to store up to 1,000 uploaded files on your account at a time, and each file must be no larger than 50 MB in size."

<http://android.stackexchange.com/questions/19092/how-can-i-read-my-epub-books-in-google-books-for-android>

So the downloaded file is now uploaded to Google Play Books where it appears to undergo some file manipulation.

I can only surmise that the file became corrupted for you during either a download or upload, or its manipulation by Google.

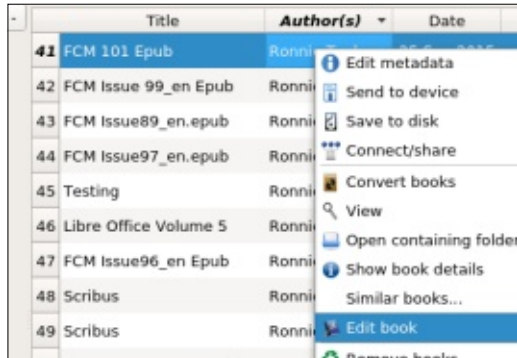
I did notice that there was one additional problem introduced by uploading the file to Google Play – whether or not it is the same under Android as in the browser I can't check right now.

Google removes the blank line between paragraphs. As paragraphs in the ePub are justified, it's difficult to see when one ends and the next begins.

If you have Calibre installed on your computer, then it is easy to edit the Metadata after importing the ePub.

Right-clicking on the ePub gives you the option to edit the book and indent the paragraphs if you

need to.



In the left column, under Styles, double-click on FCM15.css and look for

```
p {
  font-family: Ubuntu;
  font-weight: normal;
  text-align: justify;
  font-size: 12pt;
  line-height: 14pt;
}
```

and add

```
text-indent: 30pt;
```

to make it read

```
p {
  text-indent: 30pt;
  font-family: Ubuntu;
  font-weight: normal;
  text-align: justify;
  font-size: 12pt;
  line-height: 14pt;
}
```

Save and exit.

The modified ePub can now be found in the Calibre Library.

Before:



After:



Viewed from Google Books

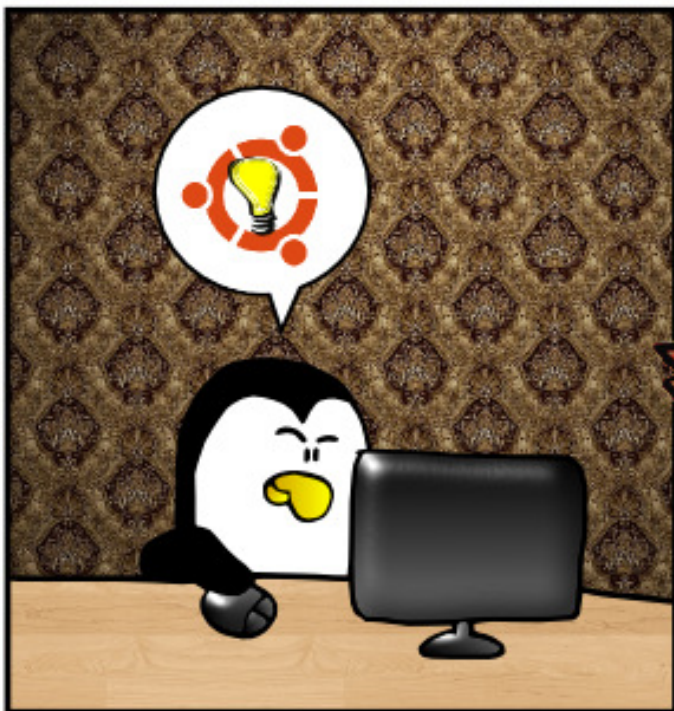
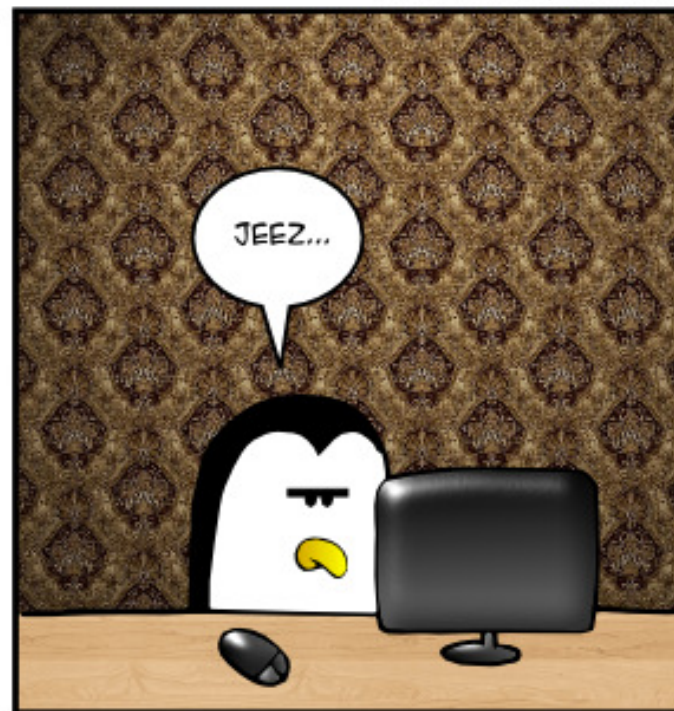
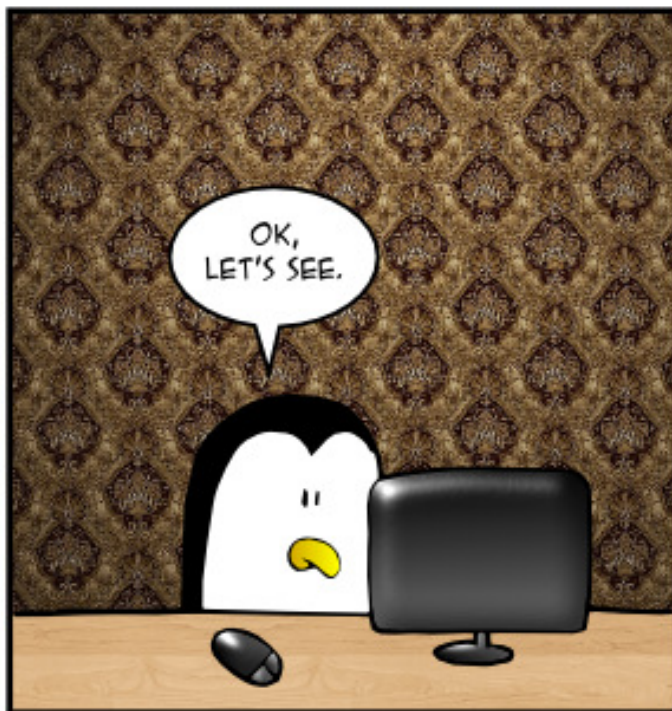
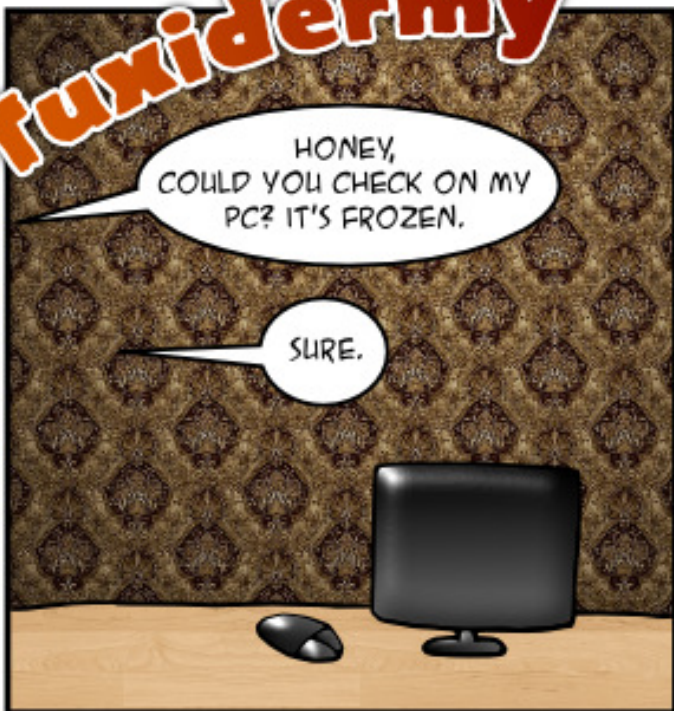
An indent of 30pt may be too much; you might be happier with, say, 12pt:

```
text-indent: 12pt;
```

Hope this is some help,

**Brian**

# Tuxidermy







# Q&A

Compiled by Gord Campbell

If you have a Linux question, email it to: [misc@fullcirclemagazine.org](mailto:misc@fullcirclemagazine.org), and Gord will answer them in a future issue. Please include as much information as you can about your query.

**Q** When I try to update my system, I get this error message:

```
Failed to fetch
http://ppa.launchpad.net/kile/stable/ubuntu/dists/vivid/main/binary-amd64/Packages 404 Not Found
```

**A** (Thanks to *claracc* in the Ubuntu Forums) You have obtained the correct response since the aforementioned ppas are not in the pointed address. You can go to software updater and in the "other software" tab, disable these ppas. Reload sources and it is fixed.

**Q** I'm planning to upgrade from 12.04 to 14.04. What should I back up?

**A** (Thanks to *mastablasta* in the Ubuntu Forums) You can make an image of the whole system (clone), or back up just individual folders. I would mostly back up just the data from my home folder for the upgrade. Then, before doing the upgrade, I

first try a live session to ensure that all works as it should.

**Q** Is there a command that I can use to get my power supply data?

**A** There is no command to query the model number, wattage rating, etc. if that is what you're asking for.

You may be able to check the voltage levels with sensors if your motherboard supports it. They're not always accurate and correctly labelled though. Install lm-sensors, then:

```
sudo sensors-detect
```

You only need to run this command once and not every time you want to see sensors.

```
sensors
```

**Q** This morning, I was notified of a software update in 14.04.

After applying the update and rebooting, I cannot sign into my system. Enter my password, get the spinning wheel and Nada. Locks up tighter than a drum after about 8 seconds. A total freeze.

**A** (Thanks to *Howefield* in the Ubuntu Forums) In the short term, try booting into a previous kernel. If you don't get a grub screen at boot up, press the shift key after switching the machine on and press the advanced options button and choose the previously working kernel.

A buggy kernel was uploaded to '-proposed' – a repository not enabled by default, so only those willing and able, with a higher tolerance to breakage, would/should be affected.

## TOP QUESTIONS AT ASKUBUNTU

\* Is there a software for visual display of disk space?  
<http://goo.gl/XZa99g>

\* Is it possible to run a Windows .msi installer?  
<http://goo.gl/SrYYmx>

\* find vs. locate  
<http://goo.gl/IgXpNQ>

\* When is it necessary to reboot an Ubuntu system?  
<http://goo.gl/NLKG3Q>

\* Timestamp, year 2038 problem for 64-bit Ubuntu system  
<http://goo.gl/NRqaPl>

\* What's the best way to write an Ubuntu ISO image on a USB stick?  
<http://goo.gl/o9H3k2>

\* How to secure my laptop so that hacking by physical access is not possible?  
<http://goo.gl/SggQ74>

\* Disk slowly filling up but no visible file size changes  
<http://goo.gl/acohCU>

\* How to execute a specific command on opening a terminal  
<http://goo.gl/6iVa7T>



## TIPS AND TECHNIQUES



## Private folder sharing

Last month I commented about the difficulties I had in setting up a server with numerous shared folders, with each one available to only a single user. I think the problem is solved.

Here are more specifics: the folders are on a drive in a USB 3.0 external dock. They are used as the target for Macrium Reflect image backups of Windows systems; a single 4 TB drive can hold image backups of all the workstations in the organization.

The key to making it work was to put an entry in /etc/fstab to mount the external drive at boot time. If you are interested, Google will reveal lots of good information about fstab. In the specific case, a new drive will be used from time to time, to allow off-site backup. The command "sudo blkid" is useful for setting up and modifying the fstab entry.

The computer is running Xubuntu, which includes the program "users and groups". For

each computer to be backed up, I added the user on the server, with the same password as on their Windows system. I also needed to set up the password in Samba. For user jean, I entered the command:

```
sudo smbpasswd -a jean
```

Then I responded to the prompts with appropriate passwords.

Then I created a folder for each user on the external drive. The last set of commands:

```
sudo nano /etc/samba/smb.conf
```

Add something along these lines:

```
[jean]
path =
/home/administrator/shares/jean
available = yes
read only = no
browseable = yes
valid users = jean
administrator
public = no
writeable = yes
directory mask = 0750
```

(save and exit)

Then:

```
sudo service smb restart
```

## testparm

And the server part is done.



**Gord** had a long career in the computer industry, then retired for several years. More recently, he somehow found himself "The IT Guy" at a 15-person accounting firm in downtown Toronto.





MEIZU

M>X4

ubuntu® edition

**SOLD OUT**

Only available in the European Union

€299.00



# BQ AQUARIS E4.5 & E5HD

Life at your fingertips

Ubuntu reinvents the way you interact with your smartphone.  
Everything you need in your day is now at your fingertips.

**AVAILABLE  
WORLDWIDE**



Let me start with a huge disclaimer. I am not a security expert. I'm a long-time computer user, and, like most people, would like to think that my activities on the Internet are, for the most part, private and secure. But privacy and security are very different from each other. Let's start with privacy.

Through history, when technology gets involved, privacy on some level is given up for the sake of convenience. As an example, here in the States, early in our history, if you wanted to get a message to someone far away, you sent a letter. Very private, but it took weeks, and sometimes months, to get that message to a person. We didn't have any technology to speed things up.

Then along came the telegraph. Now we have technology! We got our message across the lands, but first you had to hand that message to someone who then had to read it and telegraph it on until it reached its destination. Then someone on the other end wrote it down and delivered it. Less private,

but much faster. We just took it for granted that all these people were of good character and didn't disclose the content of said message to anyone who would listen.

Then along came the telephone, but not like today's phone systems. Those who had phones shared the lines with others in their area. These were known as party lines. One could easily pick up the handset and listen in on others' conversations. Again, some privacy was given up for the ability to talk to and hear the voice of a loved one far away.

Fortunately, we still have the ability to get a private message to somebody: we mail a letter. Slower, but technology affects only the speed of delivery, not the content of the message.

In the age of the Internet, we again took it for granted that when we searched the Internet or hit the send button, the email or text message we want to send just magically arrives at its destination

in mere seconds, and since it's an electronic digital message, privacy was always expected. But we were wrong. Not only are others interested in what we say, but also on what we do on the Internet. Here again, technology has reared it ugly head and privacy takes a hit.

Security, on the other hand, are the tools we use to try to make our message, the computer, and our lives, private. If we go back to that letter we sent in olden times, we might have used a wax seal with an imprint of some form to ensure that the recipient of said letter would know that it wasn't opened. Party telephone lines became private lines. Total privacy was never ensured, but we took it for granted that it was.

Now, in the Internet age, most of us go about our day taking for granted that others are watching out for us. We trust that our computers are secure because we update them regularly. We install antivirus and anti-malware software, firewalls, passwords, encryption, put locks on our doors

and blinds on our windows. These are all just the tools of security to help keep parts of our lives private. Keeping your computer updated helps keep the unwanted at bay, but most security breakdowns are usually caused by the user, not the computer or software.

So where am I going with this... you might ask. As we should all know by now, "almost" everyone and everything on the Internet wants a little piece of our privacy. From search engines to retail sites to the recently released Windows 10, everyone thinks that they know what is best for you and aims to provide it to you with every click of the mouse, wanted or not. My government, and probably yours, has a vested interest in what people are doing on or with the Internet.

What we need to ask ourselves is how much privacy are we willing to give up to use the Internet. Some would say we shouldn't have to give up any at all. Others find all this tracking to be a useful service. Is the digital highway all that much

different than a real highway? Do you care if someone sees you going to town, or only if they see you going into that establishment that only adults frequent?

If you are using a laptop with a USB stick loaded up with the Tails operating system, going from Wi-Fi spot to Wi-Fi spot in different towns, then privacy is high on your list. If you never bother with updates, and are signed up to sites like Facebook or Twitter, and have a need to post pictures of that new flat screen TV and Tweeting about going on vacation for a week starting Saturday, then privacy or security isn't a top priority.

We all view privacy differently, and conduct our lives based on that view. Do your homework, and ask questions, as you find the right balance of privacy, security and usefulness you want from your computing and phone facilities. Issues with privacy and security will be with us for some time. Don't just take them for granted, no matter how convenient they are.

## USING BASH COMMAND

### HISTORY

by Jeremy Boden

Would you like the command entry screen to be a bit less work to use? Instead of repetitive typing of commands, access to previously entered commands can be a big help. This feature works either in a "real" command entry screen (accessed via one of the CTRL+ALT+F1 thru CTRL+ALT+F6 shortcuts) or perhaps more commonly, via the Terminal window option.

It turns out that a command entry screen keeps a copy of (typically) the last 500 commands entered. One option would be to enter the history command (without any parameters), I (currently) see a long list:

```
[493 lines omitted]
494 locate amstex.sty
495 cd
...
500 sudo aptget clean
501 sudo du h /backup | sort
k 2
502 history
```

Note that the commands entered are listed in reverse order,

so that the most recent command is shown last. Actually, the history command comes with a multitude of options – which I shan't discuss. Instead, we can access the command history in a simple interactive way.

It is a "well known fact" that pressing the uparrow/downarrow keys will scroll through your command history; in particular the uparrow will display the previous command in a command screen, clearly you will want to avoid pressing this key too many times! So we search our history using "reverse intelligent search".

My command screen is waiting for me to enter a command, so it reads:

```
jeremy@hector:~$
```

Pressing CTRL+R, will cause this to change to:

```
(reversesearch)`':
```

I entered mlo – (I was expecting to type mlocate) and my screen changed to:

```
(reversesearch)`mlo': sudo
/usr/bin/updatedb.mlocate
```

• Note that the search string is shown enclosed between a

backtick character and a single quote and separated from the full command by a colon. I chose to run this command unchanged, by pressing the enter key.

- If your search string includes spaces, enter the exact number of spaces required.
- Note that in my search, it was only necessary to enter a few adjacent characters – it is not necessary to start from the beginning of the command.
- It may happen that your search string matches a number of different commands – to access an older command, just press CTRL+R again.
- To amend and run the retrieved command, press the leftarrow or rightarrow keys, type in the alterations and press enter to run the command.
- At any point, prior to pressing the enter key, you can abandon running the command by pressing CTRL+C.

After pressing enter (to run the command) or CTRL+C to cancel any command execution, your command entry screen will return to its original appearance.



# LINUX LOOPBACK

Written by S. J. Webb

Unix was developed by AT&T during the 1970's. Let's take a quick look at how AT&T started its research lab, Bell Lab. This lab fostered the growth of: radio astronomy, the transistor, the laser, information theory, the Unix OS, and the C/C++ programming language. Employees of this lab won eight Nobel Prizes for their discoveries.



Three years after Alexander Graham Bell's death, AT&T created the Bell Telephone Laboratories in 1925. It is referred to as the "Idea Factory." Over 4000 engineers and scientists from varying

departments were assigned to a new building in Murray Hill, New Jersey. This building was named the Bell Lab.

The Bell Lab core rose out of Volta Laboratory and Bureau. Volta was founded by Alexander Graham Bell. Volta Lab focused on the development of sound transmission for AT&T. Bell wanted to improve the quality of life for deaf individuals, too, from the research Volta Lab generated.

Volta laid the early ground work for Bell Lab.

Bell Labs was founded and co-owned by Western Electric and AT&T. These two companies created Bell Labs to focus solely on researching technology and equipment for the Bell Telephone Operating System. They created telephones, telephone switches, and other transmission equipment.

In the 1920's Bell Lab



demonstrated facsimile transmissions in the United States. Facsimile transmission was created in Europe earlier. The development of synchronized sound film arose, ending the silent film era. Long-distance television transmission was established by Herbert Ives to Secretary Of Commerce Herbert Hoover. One of the first encryption tools, one-time pad cipher, was developed by Gilbert Vernam and Joseph Mauborgne.

The 1930's saw the development of radio astronomy. During World War II, Bell Lab developed SIGSALY that digitally scrambled Allied speech transmissions. Also, the first photovoltaic cell was developed, which laid the groundwork for solar energy. In 1947, the first transistor was developed, which then started solid-state electronics. Additionally, Claude Shannon developed information theory, which eventually gave way to modern cryptography using various calculators.

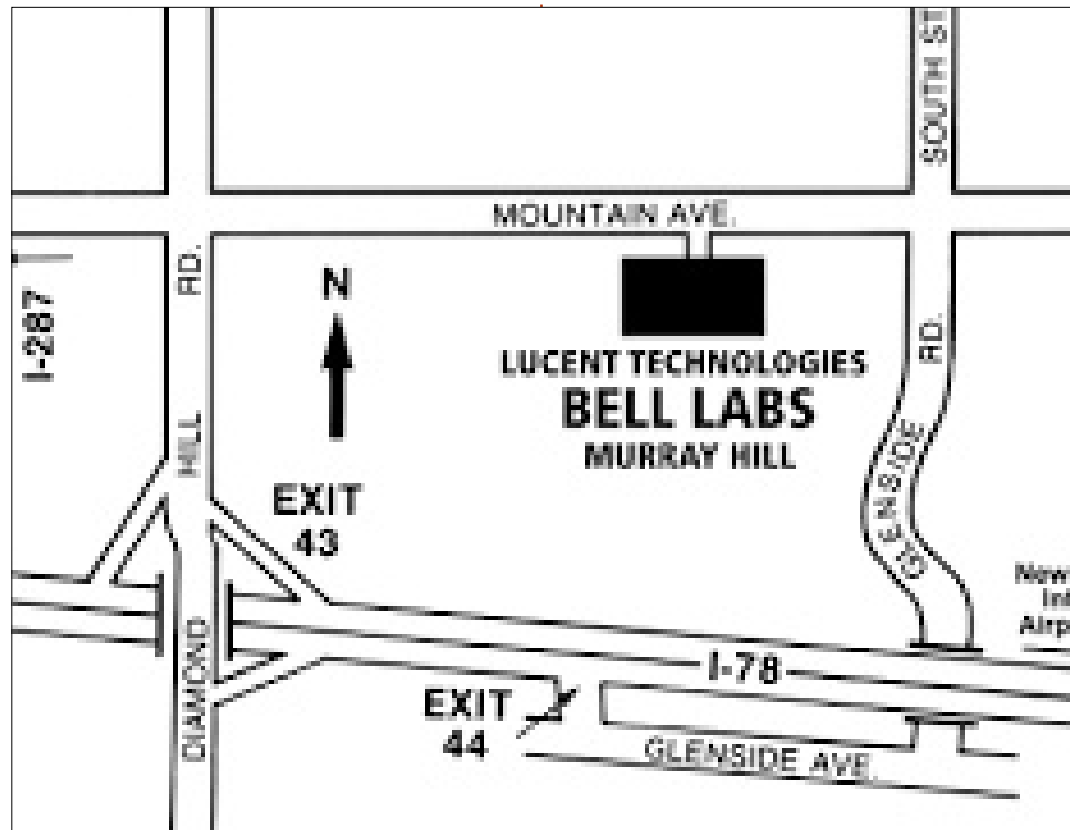
During the 1950's, the Lab

developed electronic music created by computers, and improved equipment for the Bell Telephone System. The first transatlantic phonecall between Scotland and Newfoundland was established. Computer network design thrived under Robert Prim and Joe Kruskal through contributions from their mathematical expertise. In 1958, the laser was first described in a technical paper by Art Schawlow and Charles Townes.

Next month, Linux Loopback will cover Bell Labs history from the 1960's to the present.



**SJ Webb** is a Linux Hobbyist and Research Coordinator. He enjoys fishing, hot rodding, and spending time with his kids and wife. He thanks Mike Ferarri for his mentorship.





I remember growing up last century and watching football (soccer) games on TV every weekend. My brother and I would always watch all of the important soccer games on TV with dad, so in a way it's been kind of like a family tradition to watch sports on TV (also live at a stadium, but that's a different topic). Every four years, the FIFA World Cup is a great excuse to watch games every day for a whole month. Soccer isn't the only sport we've enjoyed watching, there's also baseball, basketball, hockey, the list goes on and on.... and let's not forget the Olympics.

Growing up, I also enjoyed playing video games, but back then there was no such thing as eSports, a fancy name for competitive video games. If you had told me that one day I'd be watching other people play video games live from the comfort of my home, I wouldn't have believed it. Now, it's become common to watch video games live on a PC and to cheer for your favorite player or team.

The latest trend for eSports is

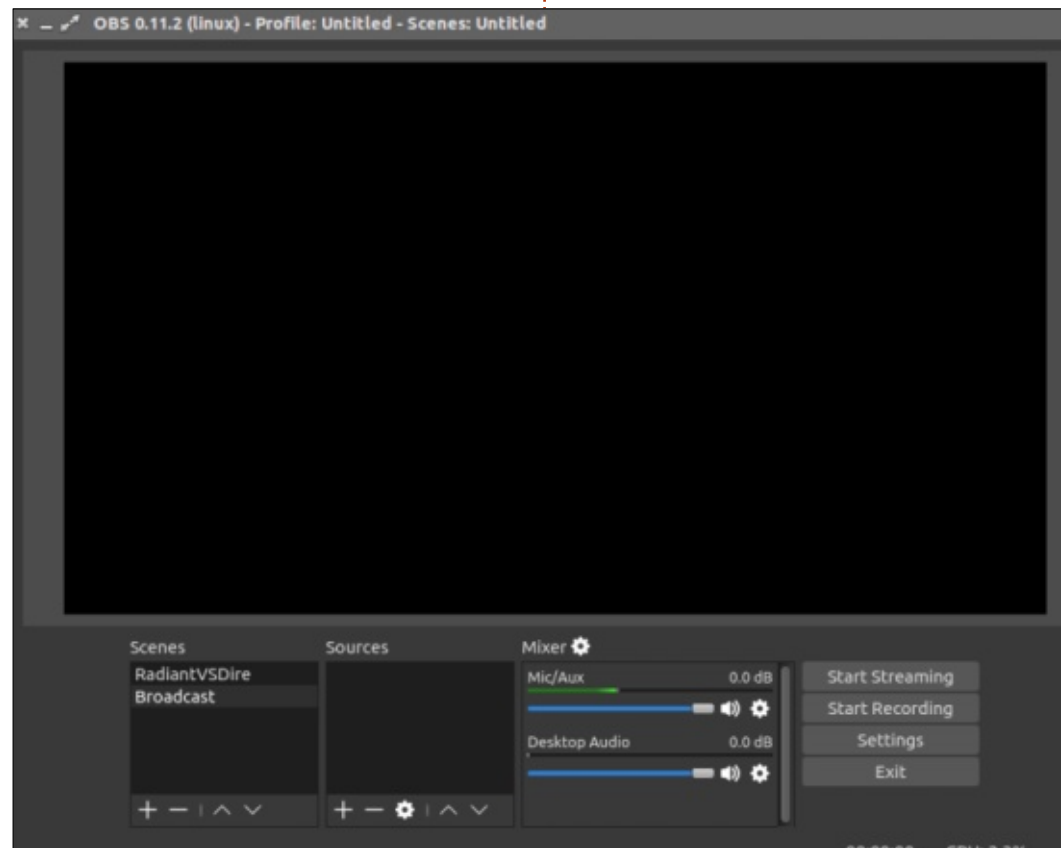
to broadcast your games while you play them, or even broadcast other people's games. Some of the top tournaments in recent eSports history have, in fact, been professionally broadcast with commentaries by broadcasters in multiple languages.

Broadcasting has taken off as

sort of a separate entity that enhances the gaming experience. In fact, there are many casters (short for broadcaster) who have developed loyal fan-bases and are broadcasting games for a living. But let's focus on you, the Linux gamer, who may be interested in broadcasting your own games for others to see. Perhaps live

broadcasting isn't your thing, but instead you would prefer to record your games and post them on Youtube or elsewhere. Whether you prefer recording or live broadcasting, Open Broadcaster Software will help you accomplish your goals.

I first found out about OBS by accident after going to <http://www.twitch.tv/> while trying to watch a professional gamer's broadcast. While browsing through the website, I came across Open Broadcaster Software and just the name alone was enough to convince me to give it a try. Ever since, I have used it for a number of things, not limited only to broadcasting but also for recording my games and for taking screenshots of games that may not support the use of screenshots in Linux. On the twitch.tv homepage, if you scroll all the way down to where it says "Become a broadcaster" and click on the 'Let's Get Started' button, you'll be taken to the twitch.tv broadcasting software downloads page. There will be five broadcasting tools





listed here, however the only one that's available for Linux is Open Broadcaster Software. Not only is it the only open source tool, it's also one of only two that you don't have to pay money to use. All of them provide a free-to-try option, but eventually they would ask you to fork over some money, except for OBS.

Instead of clicking on the Download button I suggest you click on the Setup Guide option. This will not only display a link to download the software from [obsproject.com](https://obsproject.com), but will also give you a much better set of instructions than the ones given at the [obsproject.com](https://obsproject.com) website. This is the guide that I followed when I set up OBS immediately following initial installation.

To install OBS, you can go directly to the Open Broadcaster Project website at <https://obsproject.com/> and click on the Linux option which will then take you to the Linux Download page. Here, you will find that there is an Official Ubuntu build as well as unofficial builds for Arch Linux, OpenSUSE and Gentoo, as well as the option to Build from Source. Upon downloading the official

Ubuntu build, it should automatically go through the installing process via the Ubuntu Software Center.

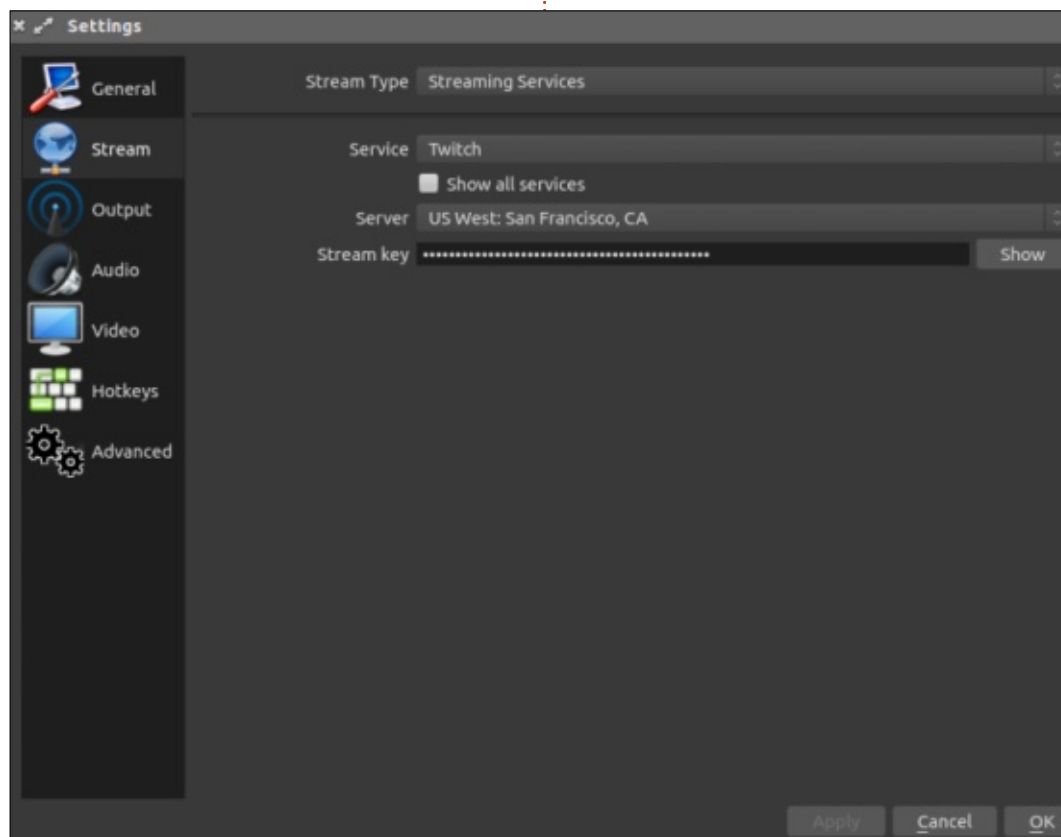
Once installed, you should take the time to adjust the settings so that it runs properly. When you first fire up OBS, you'll need to set it up, otherwise it won't be able to do much without initial user input. Rather than try to explain to you how to set it up here, instead I recommend you read either the Quick Start Guide, the Overview

Guide, or the one I mentioned from the [twitch.tv](https://www.twitch.tv) website. Those guides are pretty simple and easy to follow, while at the same time they contain all you need to get started.

Don't be overwhelmed by looking at all of the options when you first look at the OBS interface. In a nutshell, you'll find the main screen taking up the biggest real estate in the main GUI. That's where you'll see what it is that you're recording/broadcasting. On

the bottom, second from the left, you'll find a box that says Sources, which is a good place to get started and where you'll choose the source you'll be streaming/recording. Next to the Sources, in the middle, you'll find the Mixer which you may need to use from time to time. On the right are most of the important options – which are: Start Streaming, Start Recording, Settings, and Exit.

You'll first want to go to Settings and do everything that the guide requires you to do. Out of all the guides I've suggested, my favorite by far is the one provided in the [twitch.tv](https://www.twitch.tv) website. Although I've read all three of them, I've found the [twitch.tv](https://www.twitch.tv) guide the easiest to follow. If you are interested in streaming, you'll also need to create an account with [twitch.tv](https://www.twitch.tv), so that you have an outlet onto which you can stream your games. Having created your account, you'll want to go to the Dashboard on [twitch.tv](https://www.twitch.tv) and click on where it says Stream Key, then click on Show Key, and follow the prompts until your stream key is revealed. Copy the stream key, and then, under the Settings of OBS, you'll go to the Stream tab and under Service, select Twitch, then,



under Server, find the server closest to you. Finally, on the input box next to Stream Key, paste the stream key you copied earlier and you should be ready to go.

If you followed all of the instructions properly, you should now be ready to go. Now just hit the Start Streaming button on the main interface of OBS and double check on twitch.tv to make sure that it's working properly. Keep in mind that there is a slight delay in your broadcast so don't freak out if you don't see anything yet. Instead, wait a few seconds to account for the latency delay, and, if you followed the steps properly, you should see everything you have been doing a few seconds earlier on your twitch.tv channel. You're successfully broadcasting your desktop!

Now, just start up any game and it should automatically be streaming on your channel. When you're done playing, don't forget to click on the Stop Recording button. If streaming is not your thing, and instead you are interested in recording, then click on the Start Recording button to record your game (or whatever else you want to record), and,

when you're finished, click the same button which should now say Stop Recording, go to the folder that you selected to store your recordings and find your recent recording, then double-click it to watch the video and make sure everything worked fine.

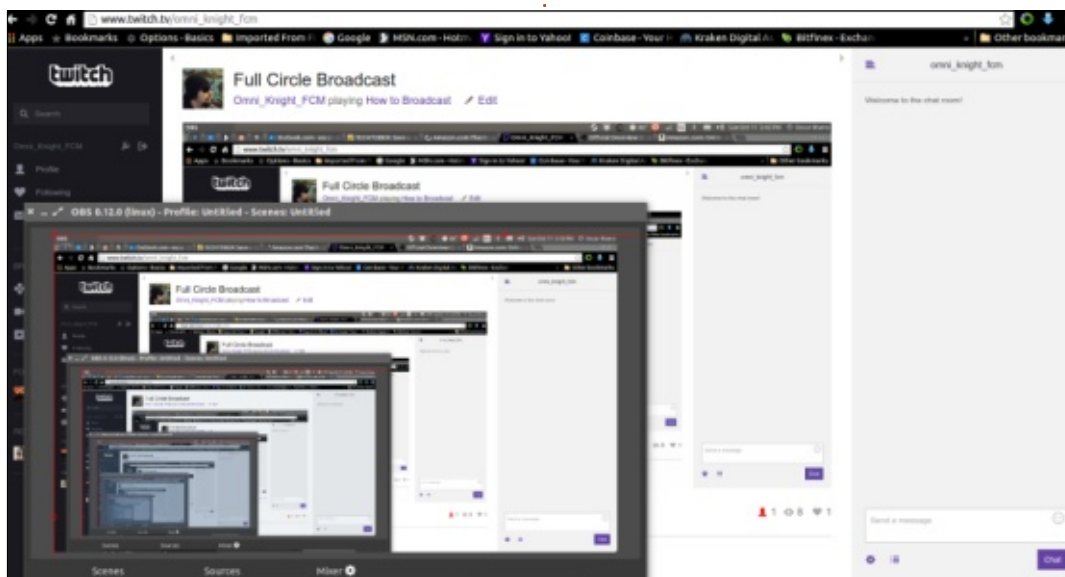
I've been using OBS for a few months now and I honestly think it's a great tool not only for gamers but also for anyone interested in recording their desktop. For example, say that you're going to make an instructional video for a friend, colleague, family member, or a stranger who may need help, then OBS is the tool that will make it very easy for you to show them exactly how to do it.

Under the Audio option, you can select to record yourself talking by choosing one of the microphones available on your PC, for example the one from your web-cam. This makes it very easy for you to explain everything while you're doing it so that, when the video is created, they can see what you've done while you also talk about it.

After using it only a couple of times I began asking myself how I managed to go so long without knowing about OBS and without using it. Go to twitch.tv if you haven't done so yet and look up any game you find to watch broadcasts of it, and, if you are so

inclined, then try out OBS and begin broadcasting or recording your own games.

I've got some exciting video game articles planned for upcoming issues, so until next time, I hope you enjoy OBS as much as I did.



**Oscar** graduated from CSUN, is a Music Director/Teacher, beta tester, Wikipedia editor, and Ubuntu Forums contributor. You can contact him via: [www.gplus.to/7bluehand](http://www.gplus.to/7bluehand) or email: [7bluehand@gmail.com](mailto:7bluehand@gmail.com)



# PATRONS

## PATRONS

Bill Berninghausen  
Jack McMahon  
Linda P  
Remke Schuurmans  
Norman Phillips  
Tom Rausner  
Charles Battersby  
Tom Bell  
Oscar Rivera  
Alex Crabtree  
John Malon  
Ray Spain  
Richard Underwood  
Charles Anderson  
Ricardo Coalla  
Chris Giltnane  
William von Hagen  
Mark Shuttleworth  
Juan Ortiz  
Joe Gulizia  
Kevin Raulins  
Doug Bruce  
Pekka Niemi  
Rob Fitzgerald  
Brian M Murray  
Roy Milner  
Brian Bogdan  
Scott Mack  
Dennis Mack  
John Helmers

## DONATIONS

John Niendorf  
Daniel Witzel  
Douglas Brown  
Donald Altman  
Patrick Scango  
Tony Wood  
Paul Miller  
Colin McCubbin  
Randy Brinson  
John Fromm  
Graham Driver  
Chris Burmajster  
Steven McKee  
Manuel Rey Garcia  
Alejandro Carmona Ligeon  
siniša vidović  
Glenn Heaton



## CHA CHA CHA CHANGES

Our admin went AWOL for months, and I had no idea if/when the site would/wouldn't get paid. Initially the plan was to move the site and domain name to my hosting, but eventually I managed to track him down and get the FCM domain name, and site hosting transferred to me.

The new site is now up. **HUGE** thanks to **Lucas Westermann** (Mr. Command & Conquer) for taking on the job of completely rebuilding the site, and scripts, from scratch, in his own time.

The Patreon page that I've set up is to help me pay the domain and hosting fees. The yearly target was quickly reached thanks to those listed on this page. FCM is not going away. Don't worry about that.

Several people have asked for a PayPal (single donation) option, so I've added a button to the side of the site

**A big thank you to all those who've used Patreon and the PayPal button. It's a big help.**

<https://www.patreon.com/fullcirclemagazine>



# HOW TO CONTRIBUTE

## FULL CIRCLE NEEDS YOU!

A magazine isn't a magazine without articles and Full Circle is no exception. We need your opinions, desktops, stories, how-to's, reviews, and anything else you want to tell your fellow \*buntu users. Send your articles to: [articles@fullcirclemagazine.org](mailto:articles@fullcirclemagazine.org)

We are always looking for new articles to include in Full Circle. For help and advice please see the **Official Full Circle Style Guide**: <http://url.fullcirclemagazine.org/75d471>

Send your **comments** or Linux experiences to: [letters@fullcirclemagazine.org](mailto:letters@fullcirclemagazine.org)  
Hardware/software **reviews** should be sent to: [reviews@fullcirclemagazine.org](mailto:reviews@fullcirclemagazine.org)  
**Questions** for Q&A should go to: [questions@fullcirclemagazine.org](mailto:questions@fullcirclemagazine.org)  
**Desktop** screens should be emailed to: [misc@fullcirclemagazine.org](mailto:misc@fullcirclemagazine.org)  
... or you can visit our **site** via: [fullcirclemagazine.org](http://fullcirclemagazine.org)



### FCM#103

**Deadline:**  
Sunday 08th Nov. 2015.  
**Release:**  
Friday 27th Nov. 2015.



## Full Circle Team



**Editor** - Ronnie Tucker  
[ronnie@fullcirclemagazine.org](mailto:ronnie@fullcirclemagazine.org)

**Webmaster** - Lucas Westermann  
[admin@fullcirclemagazine.org](mailto:admin@fullcirclemagazine.org)

**Podcast** - Les Pounder & Co.  
[podcast@fullcirclemagazine.org](mailto:podcast@fullcirclemagazine.org)

**Editing & Proofreading**  
Mike Kennedy, Gord Campbell, Robert Orsino, Josh Hertel, Bert Jerred, Jim Dyer and Emily Gonyer

Our thanks go to Canonical, the many translation teams around the world and **Thorsten Wilms** for the FCM logo.

## Getting Full Circle Magazine:



**EPUB Format** - Recent editions of Full Circle have a link to the epub file on the downloads page. If you have any problems with the epub file, you can drop an email to: [mobile@fullcirclemagazine.org](mailto:mobile@fullcirclemagazine.org)



**Issuu** - You can read Full Circle online via Issuu: <http://issuu.com/fullcirclemagazine>. Please share and rate FCM as it helps to spread the word about FCM and Ubuntu Linux.



**Google Play** - You can now read Full Circle on Google Play/Books. Either search for 'full circle magazine' or click this link: <https://play.google.com/store/books/author?id=Ronnie+Tucker>