

The PCLinuxOS magazine

Volume 115

August, 2016

Happy 25th Birthday, Linux!

***Linux 25th Birthday
Trivia & Fun Facts***

***Game Zone:
Empire Of The Gods***

***Tip Top Tips:
Kodi With The Latest Add-ons***

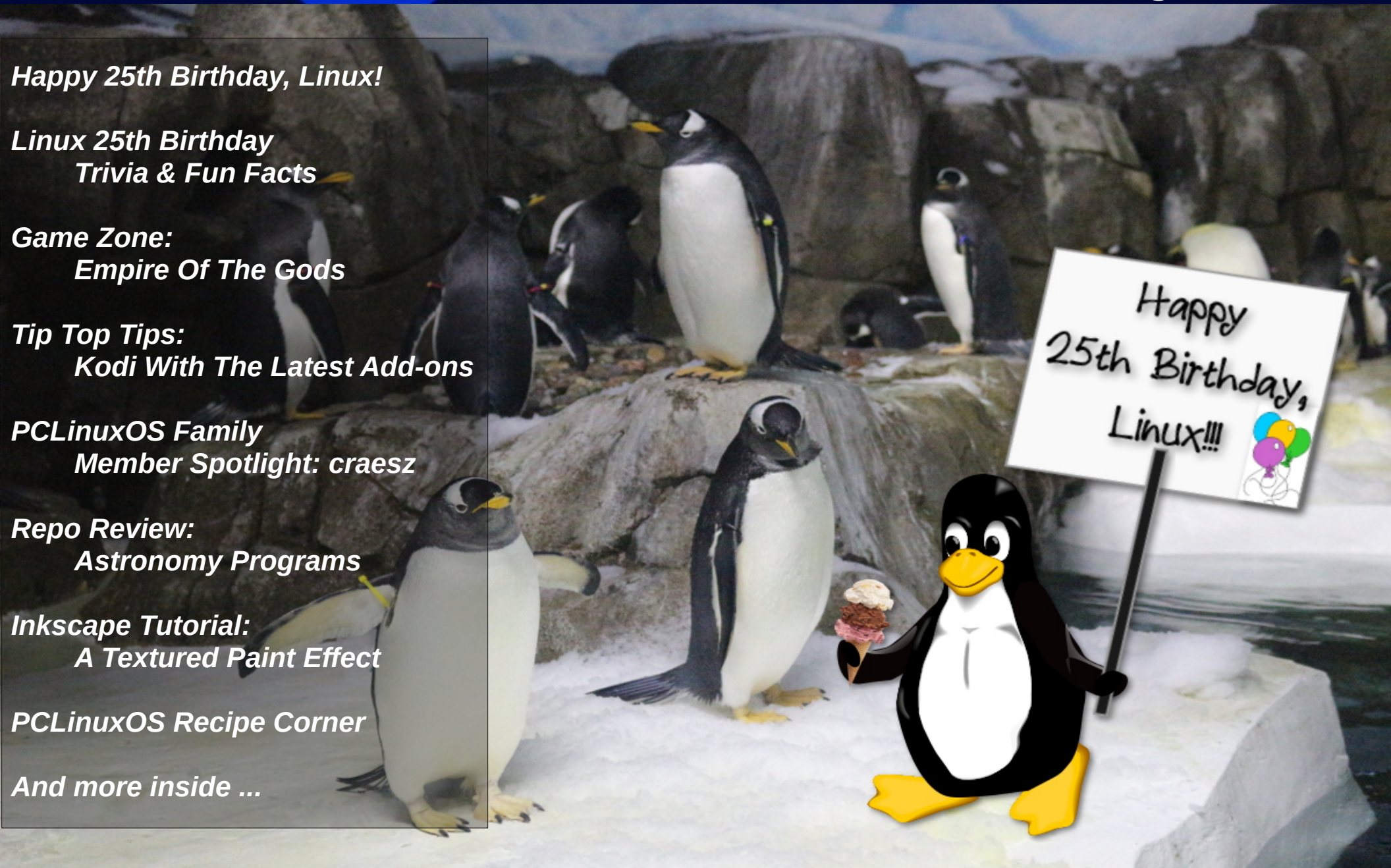
***PCLinuxOS Family
Member Spotlight: craesz***

***Repo Review:
Astronomy Programs***

***Inkscape Tutorial:
A Textured Paint Effect***

PCLinuxOS Recipe Corner

And more inside ...



In This Issue...

- 3 From The Chief Editor's Desk**
- 4 Screenshot Showcase**
- 5 Happy 25th Birthday, Linux!**
- 12 Screenshot Showcase**
- 13 Linux 25th Birthday Trivia & Fun Facts**
- 15 Screenshot Showcase**
- 16 ms_meme's Nook: Today While I Boot Up**
- 17 PCLinuxOS Recipe Corner**
- 18 Inkscape Tutorial: A Painted Text Effect**
- 19 Extending The Life Of 32 Bit Chrome On PCLinuxOS**
- 20 Screenshot Showcase**
- 21 Tip Top Tips: Kodi With The Latest Add-ons**
- 25 Repo Review: Astronomy Programs**
- 28 Screenshot Showcase**
- 29 PCLinuxOS Bonus Recipe**
- 30 PCLinuxOS Family Member Spotlight: craesz**
- 31 Game Zone: Empire Of The Gods**
- 32 Screenshot Showcase**
- 33 PCLinuxOS Puzzled Partitions**
- 36 More Screenshot Showcase**

The **PCLinuxOS** magazine

The PCLinuxOS name, logo and colors are the trademark of Texstar.

The PCLinuxOS Magazine is a monthly online publication containing PCLinuxOS-related materials. It is published primarily for members of the PCLinuxOS community. The magazine staff is comprised of volunteers from the PCLinuxOS community.

Visit us online at <http://www.pclosmag.com>

This release was made possible by the following volunteers:

Chief Editor: Paul Arnote (parnote)

Assistant Editor: Meemaw

Artwork: ms_meme, Meemaw

Magazine Layout: Paul Arnote, Meemaw, ms_meme

HTML Layout: YouCanToo

Staff:

ms_meme

Meemaw

Gary L. Ratliff, Sr.

Daniel Meiß-Wilhelm

daiaoshi

Alessandro Ebersol

loudog

YouCanToo

Pete Kelly

Antonis Komis

Khadis Thok

Smileeb

Contributors:

Cg Boy

trytip

craesz

The PCLinuxOS Magazine is released under the Creative Commons Attribution-NonCommercial-Share-Alike 3.0 Unported license. Some rights are reserved.
Copyright © 2016.



From The Chief Editor's Desk

Well, I've gone and done it. I've upgraded my first computer to 64 bit PCLinuxOS. I wish I could say that it was an easy, smooth transition, but I would be lying. I suck at lying, so I typically don't. To be honest, it was anything BUT smooth.

It started out smooth enough. It took me a few days to get Xfce setup the way I wanted that worked with my workflow. I chose to upgrade the computer I use for production of the magazine, and chose a time in the early mid-half of the month, hoping to avoid having my production machine out for the count for an extended period of time. I hoped that by doing it earlier in the month, I'd have time to fix any issues before the magazine deadline – and the end of the month magazine production crunch time – hit.



But, it didn't work out that way. Nope. No siree.

Ironically, Meemaw also upgraded one of her computers to 64 bit PCLinuxOS, too. She did hers a

little earlier in the month. But the irony doesn't end there. We both experienced a problem where neither of us could access the internet, despite our wireless indicators saying we had an outstanding, high quality connection. Plus, we experienced it at about the very same time.

Meemaw's problem was resolved by upgrading to a newer kernel, making her problem relatively short lived. When she told me about the newer kernel fixing her problem, I had high hopes that it would resolve my problem, as well. But I wasn't as lucky.

As similar as our mutual problem seemed to be, it turned out to have very different causes. My computer inexplicably "locks up." I've attributed it to a possible heat problem, although I'm not 100% sure about that. Prior to the onset of problems, the computer had "locked up" and the only way to regain access to the computer was to do a hard power reset. This may have caused some open inodes to remain, causing a possible disk problem. A reinstallation did not fix it, initially, so I reinstalled again, asking the installer to scan the drive media for bad sectors. I also stripped off all of my important files from my /home directory to a USB hard drive, so that I could just reformat the entire hard drive.

Success! It worked ... until I added one of my custom bash scripts back into the mix.

Several years ago, when I was first learning bash scripting, I wrote a program (script) called "Touchpad Toggle." It's simple enough. It simply toggles the touchpad on or off. A modified version of my original script is in the PCLinuxOS repository, as – you guessed it – "Touchpad Toggle."

A few months ago, forum member **The Chief** was seeking a solution that automatically

enabled/disabled the touchpad, based on whether or not a USB mouse was connected to the laptop. It caused me to revisit Touchpad Toggle, because what he was asking for was what my ultimate goal was. With a few more years of bash scripting under my belt (I'm nowhere near expert, though), coming up with the [solution](#) was much easier than I anticipated. It worked perfectly, except for one thing: it took out my connection to the internet. However, it worked perfectly for The Chief and others. So, some unique state must exist between my particular combination of hardware and the notification area that causes my internet access to disappear whenever a bash script messes with the notification area or pops up a notification. The connection between the hardware, internet connection and notification area is so *weird* that I cannot even fathom an explanation as to why my internet disappears.

I even tried using the modified version of Touchpad Toggle from the repos, but it also behaved similarly, knocking out my internet access. All because the modification placed an icon in the notification area of the panel. The repo version of Touchpad Toggle works perfectly for Meemaw on her computer, albeit on a computer with a much different hardware profile. The **ONLY** version of Touchpad Toggle that worked on this particular computer is my original version, which is a version that does nothing at all to the notification area. Instead, it uses a dialog box that appears on the screen to alert you when the touchpad state has been toggled.

Now that I finally have everything working as it should, I'm a bit gun shy about trying anything that might upset the apple cart again, if you know what I mean. But then, I am curious to explore the issue further to see if I can pin down the culprit that knocks my internet out. I wonder if it's `net_applet`, since

Meemaw has switched to using wicd in place of net_applet. But others use net_applet with my touchpad-toggle-auto.sh script without issue. The touchpad-toggle-auto.sh script also has no deleterious effects when used with a wired internet connection, even with net_applet managing the wired connection. It also makes me curious if it's a peculiarity with the ralink driver for the wireless adapter on this particular computer. Maybe it's the call to the command that returns the status of and alters the status of the touchpad, because even my "safe" version of the Touchpad Toggle script will knock me offline from time to time. And that ONE command is what is common between the four versions of the script that I've tried. These are the types of things that drive programmer types insane while trying to figure them out.

My experience upgrading this one laptop to 64 bit has also made me somewhat reluctant to upgrade my other capable computers to the 64 bit version of PCLinuxOS. I know that I will eventually have to do so, but it does cause some consternation and reluctance. Coming fresh off of this endeavor, the others can wait a while to be upgraded.

So, until next month, I bid you peace, happiness, serenity and prosperity ... and sanity!



CHIMPBOX

Big Things, Little Packages. Measuring only 7.5 X 8.5 X 2 inches
<http://chimpbox.us>

Screenshot Showcase



Posted by weirdwolf, July 15, 2016, running LXDE.

Happy 25th Birthday, Linux!

by Paul Arnote (parnote)

As hard as it may be to believe, Linux turns 25 years old later this month. It's easy for me to remember, because Linus Benedict Torvalds announced Linux to the world on my birthday, August 25.



Enrolled in the University of Helsinki, a young Linus Torvalds had gotten his hands on a 386 computer – state of the art in its day. It was Intel's first 32 bit processor, and he wanted to be able to unlock its potential. There was a Unix operating system available for free, but only for educational purposes. It was called Minix. Its creator would not allow its source code to be altered, and largely ignored user requests for features. Minix featured, among other things, a modular kernel, in the belief that it would be easier to maintain. Unfortunately, it was only a 16 bit design, and its creator was reticent to make a 32 bit version. All other Unix systems available for the new 32 bit platform were prohibitively expensive for regular, individual users.

Thus, Linus Torvalds set out to make his own free kernel. At first, he built Linux on a computer running Minix, but insured that Linux was free of proprietary Minix code. The rest of the story has been told and retold over the years, and is easily found on the internet.

So, for Linux's 25th birthday, I want to do something slightly different. About a year after releasing Linux to the public, Linus Torvalds gathered up many of the Usenet posts and correspondence from the time around Linux's release. What follows is Linus Torvald's account of Linux History, in the form of the communications that took place at the time.

LINUX's History

Note: The following text was written by Linus on July 31 1992. It is a collection of various artifacts from the period in which Linux first began to take shape.

This is just a sentimental journey into some of the first posts concerning linux, so you can happily press 'n' now if you actually thought you'd get anything technical.

From: torvalds@klaava.Helsinki.FI (Linus Benedict Torvalds)
Newsgroups: comp.os.minix
Subject: Gcc-1.40 and a posix-question
Message-ID:
Date: 3 Jul 91 10:00:50 GMT

Hello netlanders,

Due to a project I'm working on (in minix), I'm interested in the posix standard definition. Could somebody please point me to a (preferably) machine-readable format of the latest posix rules? Ftp-sites would be nice.

The project was obviously linux, so by July 3rd I had started to think about actual user-level things: some of the device drivers were ready, and the harddisk actually worked. Not too much else.

As an aside for all using gcc on minix - [deleted]

Just a success-report on porting gcc-1.40 to minix using the 1.37 version made by Alan W Black & co.

Linus Torvalds torvalds@kruuna.helsinki.fi

PS. Could someone please try to finger me from overseas, as I've installed a "changing .plan" (made by your's truly), and I'm not certain it works from outside? It should report a new .plan every time.

So I was clueless - had just learned about named pipes. Sue me. This part of the post got a lot more response than the actual POSIX query, but the query did lure out arl from the woodwork, and we mailed around for a bit, resulting in the Linux subdirectory on nic.funet.fi.

Then, almost two months later, I actually had something working: I made sources for version 0.01 available on nic sometimes around this time. 0.01 sources weren't actually runnable: they were just a token gesture to arl who had probably started to despair about ever getting anything. This next post must have been from just a couple of weeks before that release.

From: torvalds@klaava.Helsinki.FI (Linus Benedict Torvalds)
Newsgroups: comp.os.minix
Subject: What would you like to see most in minix?
Summary: small poll for my new operating system
Message-ID:
Date: 25 Aug 91 20:57:08 GMT
Organization: University of Helsinki

Hello everybody out there using minix -

I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones. This has been brewing since april, and is starting to get ready. I'd like any feedback on things people like/dislike in minix, as my OS resembles it somewhat (same physical layout of the file-system (due to practical reasons) among other things).

I've currently ported bash(1.08) and gcc(1.40), and things seem to work. This implies that I'll get something practical within a few months, and I'd like to know what features most people would want. Any suggestions are welcome, but I won't promise I'll implement them :-)

Linus (torvalds@kruuna.helsinki.fi)

PS. Yes - it's free of any minix code, and it has a multi-threaded fs. It is NOT protable (uses 386 task switching etc), and it probably never will support anything other than AT-harddisks, as that's all I have :-).

Judging from the post, 0.01 wasn't actually out yet, but it's close. I'd guess the first version went out in the middle of September -91. I got some responses to this (most by mail, which I haven't saved), and I even got a few mails asking to be beta-testers for linux. After that just a few general answers to questions on the net:

From: torvalds@klaava.Helsinki.FI (Linus Benedict Torvalds)
Newsgroups: comp.os.minix
Subject: Re: What would you like to see most in minix?
Summary: yes - it's nonportable
Message-ID:
Date: 26 Aug 91 11:06:02 GMT
Organization: University of Helsinki

In article jkp@cs.HUT.FI (Jyrki Kuoppala) writes:
>> [re: my post about my new OS]
>
> Tell us more! Does it need a MMU?

Yes, it needs a MMU (sorry everybody), and it specifically needs a 386/486 MMU (see later).

>
>> PS. Yes - it's free of any minix code, and it has a multi-threaded fs.
>> It is NOT protable (uses 386 task switching etc)
>
> How much of it is in C? What difficulties will there be in porting?
> Nobody will believe you about non-portability ;-), and I for one would
> like to port it to my Amiga (Mach needs a MMU and Minix is not free).

Simply, I'd say that porting is impossible. It's mostly in C, but most people wouldn't call what I write C. It uses every conceivable feature of the 386 I could find, as it was also a project to teach me about the 386. As already mentioned, it uses a MMU, for both paging (not to disk yet) and segmentation. It's the segmentation that makes it REALLY 386 dependent (every task has a 64Mb segment for code & data - max 64 tasks in 4Gb. Anybody who needs more than 64Mb/task - tough cookies).

It also uses every feature of gcc I could find, specifically the `__asm__` directive, so that I wouldn't need so much assembly language objects. Some of my "C"-files (specifically mm.c) are almost as much assembler as C. It would be "interesting" even to port it to another compiler (though why anybody would want to use anything other than gcc is a mystery).

Note: linux has in fact gotten more portable with newer versions: there was a lot more assembly in the early versions. It has in fact been ported to other architectures by now.

Unlike minix, I also happen to LIKE interrupts, so interrupts are handled without trying to hide the reason behind them (I especially like my hard-disk-driver. Anybody else make interrupts drive a state-machine?). All in all it's a porters nightmare.

```
>As for the features; well, pseudo ttys, BSD sockets, user-mode
>filesystems (so I can say cat /dev/tcp/kruuna.helsinki.fi/
finger),
>window size in the tty structure, system calls capable of
supporting
>POSIX.1.0h, and bsd-style long file names.
```

Most of these seem possible (the tty structure already has stubs for window size), except maybe for the user-mode filesystems. As to POSIX, I'd be delighted to have it, but posix wants money for their papers, so that's not currently an option. In any case these are things that won't be supported for some time yet (first I'll make it a simple minix-lookalike, keyword SIMPLE).

Linus (torvalds@kruuna.helsinki.fi)

PS. To make things really clear - yes I can run gcc on it, and bash, and most of the gnu [bin/file]utilities, but it's not very debugged, and the library is really minimal. It doesn't even support floppy-disks yet. It won't be ready for distribution for a couple of months. Even then it probably won't be able to do much more than minix, and much less in some respects. It will be free though (probably under gnu-license or similar).



Well, obviously something worked on my machine: I doubt I had yet gotten gcc to compile itself under linux (or I would have been too proud of it not to mention it). Still before any release-date.

Then, October 5th, I seem to have released 0.02. As I already mentioned, 0.01 didn't actually come with any binaries: it was just source code for people interested in what linux looked like. Note the lack of announcement for 0.01: I wasn't too proud of it, so I think I only sent a note to everybody who had shown interest.

From: torvalds@klaava.helsinki.fi (Linus Benedict Torvalds)
Newsgroups: comp.os.minix
Subject: Free minix-like kernel sources for 386-AT
Message-ID:
Date: 5 Oct 91 05:41:06 GMT
Organization: University of Helsinki

Do you pine for the nice days of minix-1.1, when men were men and wrote their own device drivers? Are you without a nice project and just dying to cut your teeth on a OS you can try to modify for your needs? Are you finding it frustrating when everything works on minix? No more all-nighters to get a nifty program working? Then this post might be just for you :-)

As I mentioned a month(?) ago, I'm working on a free version of a minix-lookalike for AT-386 computers. It has finally reached the stage where it's even usable (though may not be depending on what you want), and I am willing to put out the sources for wider distribution. It is just version 0.02 (+1 (very small) patch already), but I've successfully run bash/gcc/gnu-make/gnu-sed/compress etc under it.

Sources for this pet project of mine can be found at nic.funet.fi (128.214.6.100) in the directory /pub/OS/Linux. The directory also contains some README-file and a couple of binaries to work under linux (bash, update and gcc, what more can you ask for :-). Full kernel source is provided, as no minix code has been used. Library sources are only partially free, so that cannot be distributed currently. The system is able to compile "as-is" and has been known to work. Heh. Sources to the binaries (bash and gcc) can be found at the same place in /pub/gnu.

ALERT! WARNING! NOTE! These sources still need minix-386 to be compiled (and gcc-1.40, possibly 1.37.1, haven't tested), and you need minix to set it up if you want to run it, so it is not yet a standalone system for those of you without minix. I'm working on it. You also need to be something of a hacker to set it up (?), so for those hoping for an alternative to minix-386, please ignore me. It is currently meant for hackers interested in operating

systems and 386's with access to minix.

The system needs an AT-compatible harddisk (IDE is fine) and EGA/VGA. If you are still interested, please ftp the README/RELNOTES, and/or mail me for additional info.

I can (well, almost) hear you asking yourselves "why?". Hurd will be out in a year (or two, or next month, who knows), and I've already got minix. This is a program for hackers by a hacker. I've enjoyed doing it, and somebody might enjoy looking at it and even modifying it for their own needs. It is still small enough to understand, use and modify, and I'm looking forward to any comments you might have.

I'm also interested in hearing from anybody who has written any of the utilities/library functions for minix. If your efforts are freely distributable (under copyright or even public domain), I'd like to hear from you, so I can add them to the system. I'm using Earl Chews estdio right now (thanks for a nice and working system Earl), and similar works will be very welcome. Your (C)'s will of course be left intact. Drop me a line if you are willing to let me use your code.

Linus

PS. to PHIL NELSON! I'm unable to get through to you, and keep getting "forward error - strawberry unknown domain" or something.

Well, it doesn't sound like much of a system, does it? It did work, and some people even tried it out. There were several bad bugs (and there was no floppy-driver, no VM, no nothing), and 0.02 wasn't really very useable.

0.03 got released shortly thereafter (max 2-3 weeks was the time between releases even back then), and 0.03 was pretty useable. The next version was numbered 0.10, as things actually started to work pretty well. The next post gives some idea of what had happened in two months more...

From: torvalds@klaava.Helsinki.FI (Linus Benedict Torvalds)
Newsgroups: comp.os.minix
Subject: Re: Status of LINUX?
Summary: Still in beta
Message-ID:
Date: 19 Dec 91 23:35:45 GMT
Organization: University of Helsinki

In article miquels@maestro.htsa.aha.nl (Miquel van Smoorenburg) writes:
>Hello *,
>I know some people are working on a FREE O/S for the 386/486,

>under the name Linux. I checked nic.funet.fi now and then, to see what was
>happening. However, for the time being I am without FTP access so I don't
>know what is going on at the moment. Could someone please inform me about it?
>It's maybe best to follow up to this article, as I think that there are
>a lot of potential interested people reading this group. Note, that I don't
>really *have* a >= 386, but I'm sure in time I will.

Linux is still in beta (although available for brave souls by ftp), and has reached the version 0.11. It's still not as comprehensive as 386-minix, but better in some respects. The "Linux info-sheet" should be posted here some day by the person that keeps that up to date. In the meantime, I'll give some small pointers.

First the bad news:

- Still no SCSI: people are working on that, but no date yet. Thus you need a AT-interface disk (I have one report that it works on an EISA 486 with a SCSI disk that emulates the AT-interface, but that's more of a fluke than anything else: ISA+AT-disk is currently the hardware setup)

As you can see, 0.11 had already a small following. It wasn't much, but it did work.

- still no init/login: you get into bash as root upon bootup.

That was still standard in the next release.

- although I have a somewhat working VM (paging to disk), it's not ready yet. Thus linux needs at least 4M to be able to run the GNU binaries (especially gcc). It boots up in 2M, but you cannot compile.

I actually released a 0.11+VM version just before Christmas -91: I didn't need it myself, but people were trying to compile the kernel in 2MB and failing, so I had to implement it. The 0.11+VM version was available only to a small number of people that wanted to test it out: I'm still surprised it worked as well as it did.

- minix still has a lot more users: better support.

- it hasn't got years of testing by thousands of people, so there are probably quite a few bugs yet.

Then for the good things..

- It's free (copyright by me, but freely distributable under a very lenient copyright)

The early copyright was in fact much more restrictive than the GNU copyleft: I didn't allow any money at all to change hands due to linux. That changed with 0.12.

- it's fun to hack on.

- /real/ multithreading filesystem.

- uses the 386-features. Thus locked into the 386/486 family, but it makes things clearer when you don't have to cater to other chips.

- a lot more... read my .plan.

/I/ think it's better than minix, but I'm a bit prejudiced. It will never be the kind of professional OS that Hurd will be (in the next century or so :), but it's a nice learning tool (even more so than minix, IMHO), and it was/is fun working on it.

Linus (torvalds@kruuna.helsinki.fi)

---- my .plan -----
Free UNIX for the 386 - coming 4QR 91 or 1QR 92.

The current version of linux is 0.11 - it has most things a unix kernel needs, and will probably be released as 1.0 as soon as it gets a little more testing, and we can get a init/login going. Currently you get dumped into a shell as root upon bootup.

Linux can be gotten by anonymous ftp from 'nic.funet.fi' (128.214.6.100) in the directory '/pub/OS/Linux'. The same directory also contains some binary files to run under Linux. Currently gcc, bash, update, uemacs, tar, make and fileutils. Several people have gotten a running system, but it's still a hackers kernel.

Linux still requires a AT-compatible disk to be useful: people are working on a SCSI-driver, but I don't know when it will be ready.

There are now a couple of other sites containing linux, as people have had difficulties with connecting to nic. The sites are:

Tupac-Amaru.Informatik.RWTH-Aachen.DE (137.226.112.31):

directory /pub/msdos/replace

tsx-11.mit.edu (18.172.1.2):
directory /pub/linux

There is also a mailing list set up 'Linux-activists@niksula.hut.fi'. To join, mail a request to 'Linux-activists-request@niksula.hut.fi'. It's no use mailing me: I have no actual contact with the mailing-list (other than being on it, naturally).

Mail me for more info:

Linus (torvalds@kruuna.helsinki.fi)

0.11 has these new things:

- demand loading
- code/data sharing between unrelated processes
- much better floppy drivers (they actually work mostly)
- bug-corrections
- support for Hercules/MDA/CGA/EGA/VGA
- the console also beeps (Wow! Wonder-kernel :-)
- mkfs/fsck/fdisk
- US/German/French/Finnish keyboards
- settable line-speeds for com1/2



As you can see: 0.11 was actually stand-alone: I wrote the first mkfs/fsck/fdisk programs for it, so that you didn't need minix any more to set it up. Also, serial lines had been hard-coded to 2400bps, as that was all I had.

Still lacking:

- init/login
- rename system call
- named pipes
- symbolic links

Well, they are all there now: init/login didn't quite make it to 0.12, and rename() was implemented as a patch somewhere between 0.12 and 0.95. Symlinks were in 0.95, but named pipes didn't make it until 0.96.

Note: The version number went directly from 0.12 to 0.95, as the follow-on to 0.12 was getting feature-full enough to deserve a number in the 0.90's

0.12 will probably be out in January (15th or so), and will have:

- POSIX job control (by tytso)
- VM (paging to disk)
- Minor corrections

Actually, 0.12 was out January 5th, and contained major corrections. It was in fact a very stable kernel: it worked on a lot of new hardware, and there was no need for patches for a long time. 0.12 was also the kernel that "made it": that's when Linux started to spread a lot faster. Earlier kernel releases were very much only for hackers: 0.12 actually worked quite well.

Note: The following document is a reply by Linus Torvalds, creator of Linux, in which he talks about his experiences in the early stages of Linux development

To: Linux-Activists@BLOOM-PICAYUNE.MIT.EDU
From: torvalds@klaava.Helsinki.FI (Linus Benedict Torvalds)
Subject: Re: Writing an OS - questions !!
Date: 5 May 92 07:58:17 GMT

In articlenani@td2cad.intel.com (V. Narayanan) writes:

Hi folks,

For quite some time this "novice" has been wondering as to how one goes about the task of writing an OS from "scratch". So here are some questions, and I would appreciate if you could take time to answer 'em.

Well, I see someone else already answered, but I thought I'd take on the linux-specific parts. Just my personal experiences, and I don't know how normal those are.

1) How would you typically debug the kernel during the development phase?

Depends on both the machine and how far you have gotten on the kernel: on more simple systems it's generally easier to set up. Here's what I had to do on a 386 in protected mode.

The worst part is starting off: after you have even a minimal system you can use printf etc, but moving to protected mode on a 386 isn't fun, especially if you at first don't know the architecture very well. It's distressingly easy to reboot the system at this stage: if the 386 notices something is wrong, it

shuts down and reboots - you don't even get a chance to see what's wrong.

Printf() isn't very useful - a reboot also clears the screen, and anyway, you have to have access to video-mem, which might fail if your segments are incorrect etc. Don't even think about debuggers: no debugger I know of can follow a 386 into protected mode. A 386 emulator might do the job, or some heavy hardware, but that isn't usually feasible.

What I used was a simple killing-loop: I put in statements like

```
die: jmp die
```

at strategic places. If it locked up, you were ok, if it rebooted, you knew at least it happened before the die-loop. Alternatively, you might use the sound io ports for some sound-clues, but as I had no experience with PC hardware, I didn't even use that. I'm not saying this is the only way: I didn't start off to write a kernel, I just wanted to explore the 386 task-switching primitives etc, and that's how I started off (in about April-91).

After you have a minimal system up and can use the screen for output, it gets a bit easier, but that's when you have to enable interrupts. Bang, instant reboot, and back to the old way. All in all, it took about 2 months for me to get all the 386 things pretty well sorted out so that I no longer had to count on avoiding rebooting at once, and having the basic things set up (paging, timer-interrupt and a simple task-switcher to test out the segments etc).

2) Can you test the kernel functionality by running it as a process on a different OS? Wouldn't the OS(the development environment) generate exceptions in cases when the kernel (of the new OS) tries to modify 'privileged' registers?

Yes, it's generally possible for some things, but eg device drivers usually have to be tested out on the bare machine. I used minix to develop linux, so I had no access to IO registers, interrupts etc. Under DOS it would have been possible to get access to all these, but then you don't have 32-bit mode. Intel isn't that great - it would probably have been much easier on a 68040 or similar.

So after getting a simple task-switcher (it switched between two processes that printed AAAA...and BBBB...respectively by using the timer-interrupt - Gods I was proud over that), I still had to continue debugging basically by using printf. The first thing written was the keyboard driver: that's the reason it's still written completely in assembler (I didn't dare move to C yet - I

was still debugging at about instruction-level).

After that I wrote the serial drivers, and voila, I had a simple terminal program running (well, not that simple actually). It was still the same two processes (AAA..), but now they read and wrote to the console/serial lines instead. I had to reboot to get out of it all, but it was a simple kernel.

After that it was plain sailing: hairy coding still, but I had some devices, and debugging was easier. I started using C at this stage, and it certainly speeds up development. This is also when I start to get serious about my megalomaniac ideas to make "a better minix that minix". I was hoping I'd be able to recompile gcc under linux some day...

The harddisk driver was more of the same: this time the problems with bad documentation started to crop up. The PC may be the most used architecture in the world right now, but that doesn't mean the docs are any better: in fact I haven't seen /any/ book even mentioning the weird 386-387 coupling in an AT etc (Thanks Bruce).

After that, a small filesystem, and voila, you have a minimal unix. Two months for basic setups, but then only slightly longer until I had a disk-driver (seriously buggy, but it happened to work on my machine) and a small filesystem. That was about when I made 0.01 available (late august-91? Something like that): it wasn't pretty, it had no floppy driver, and it couldn't do much anything. I don't think anybody ever compiled that version. But by then I was hooked, and didn't want to stop until I could chuck out minix.

3) Would new linkers and loaders have to be written before you get a basic kernel running?

All versions up to about 0.11 were cross-compiled under minix386 - as were the user programs. I got bash and gcc eventually working under 0.02, and while a race-condition in the buffer-cache code prevented me from recompiling gcc with itself, I was able to tackle smaller compiles. 0.03 (October?) was able to recompile gcc under itself, and I think that's the first version that anybody else actually used. Still no floppies, but most of the basic things worked.

After 0.03 I decided that the next version was actually useable (it was, kind of, but boy is X under 0.96 more impressive), and I called the next version 0.10 (November?). It still had a rather serious bug in the buffer-cache handling code, but after patching that, it was pretty ok. 0.11 (December) had the first floppy driver, and was the point where I started doing linux development

The PCLinuxOS Magazine Special Editions!

Get Your Free Copies Today!

under itself. Quite as well, as I trashed my minix386 partition by mistake when trying to autodial /dev/hd2.

By that time others were actually using linux, and running out of memory. Especially sad was the fact that gcc wouldn't work on a 2MB machine, and although c386 was ported, it didn't do everything gcc did, and couldn't recompile the kernel. So I had to implement disk-paging: 0.12 came out in January (?) and had paging by me as well as job control by tytso (and other patches: pmacdona had started on VC's etc). It was the first release that started to have "non-essential" features, and being partly written by others. It was also the first release that actually did many things better than minix, and by now people started to really get interested.

Then it was 0.95 in March, bugfixes in April, and soon 0.96. It's certainly been fun (and I trust will continue to be so) - reactions have been mostly very positive, and you do learn a lot doing this type of thing (on the other hand, your studies suffer in other respects :)

Linus



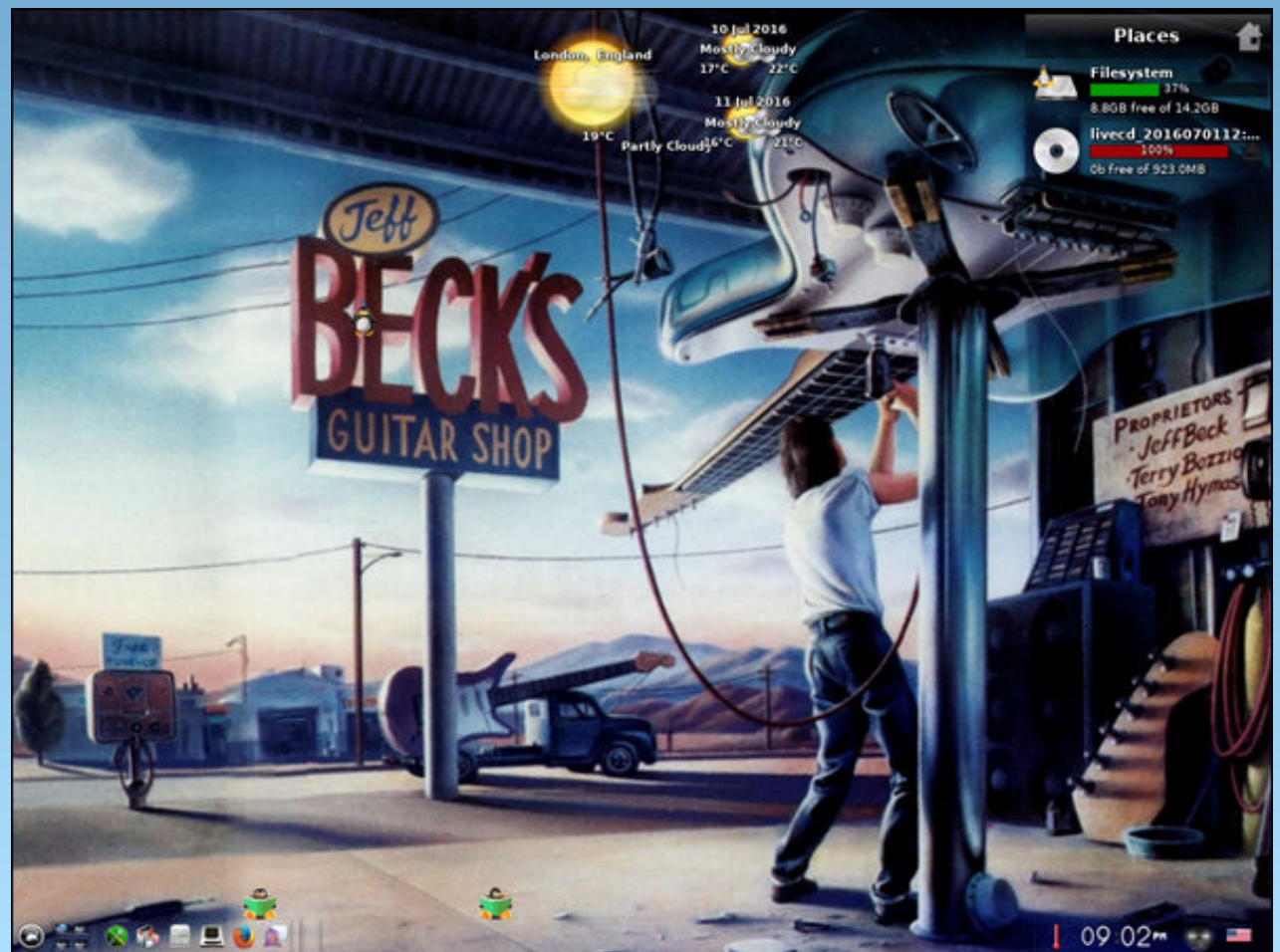
*Looking for an old article?
Can't find what you want? Try the*

**PCLinuxOS Magazine's
searchable index!**

The **PCLinuxOS** magazine



Screenshot Showcase



Posted by OnlyHuman, July 10, 2016, running e21.

Linux 25th Birthday Trivia & Fun Facts

by Paul Arnote (parnote)

Here are some interesting facts and tidbits about our favorite operating system, certain to tickle the curiosity of even the most casual Linux user.

The original name wasn't what we know today. Linux had been previously considered and rejected by Linus Torvalds as possibly being seen as too egotistical. The original name was to be Freax, formed by the words free and freak (the latter an homage to the name applied to computer nerds of the day), with the "x" alluding to Unix.

The name change from Freax to Linux came about when Ari Lemmke, a coworker of Linus Torvalds at the Helsinki University of Technology and a volunteer administrator of the FTP server, didn't think "Freax" was a good name. So, without consulting Torvalds, he changed the name of the project to "Linux" on the server. Linus later consented to the name change.

The first release of the Linux kernel, 0.01, contained 10,239 lines of code.

The latest Linux kernel has over 19,000,000 (yes, million) lines of code, contributed by over 4,000 corporations, individuals and maintainers. But not all of those lines are compiled into the kernel. Some sections are old and unused, others exist to facilitate portability, some are experimental, and others just aren't needed for every system.



The Linux kernel source code, being as large as it is, has its share of "bad words." The latest count, which goes up through kernel 4.5.4, kind of goes like this. There are 179 incidences of a word that refers to human excrement (rhymes with map), and 136 that refer to another word for human excrement (rhymes with bit). There are 41 occurrences of another word referring to coitus (rhymes with truck), 24 occurrences of another word that refers to an illegitimate male child (rhymes with mastered), and six occurrences of another word that refers to urination (rhymes with miss).

Similarly, the Linux kernel has its share of "good words." Once again, the latest count goes up

through kernel 4.5.4. There are 4,332 occurrences of the word "good" in the Linux kernel. The word "nice" shows up 1,363 times. The word "love" occurs 658 times, while "kiss" appears 229 times, and "sweet" appears 71 times.

Linux isn't just developed by aging nerds hacking away in their parent's basement. Over 80% of Linux development comes from paid developers working for a number of corporate entities.

Some of those corporate contributors to Linux development include some surprises. Microsoft has even contributed to Linux development, to help

insure that Linux users are able to utilize Azure and Hyper-V efficiently. Others top corporate contributors include Red Hat, Suse, Novell, Oracle, Linaro, Samsung, HP, Broadcom, IBM, and Intel among the mix.

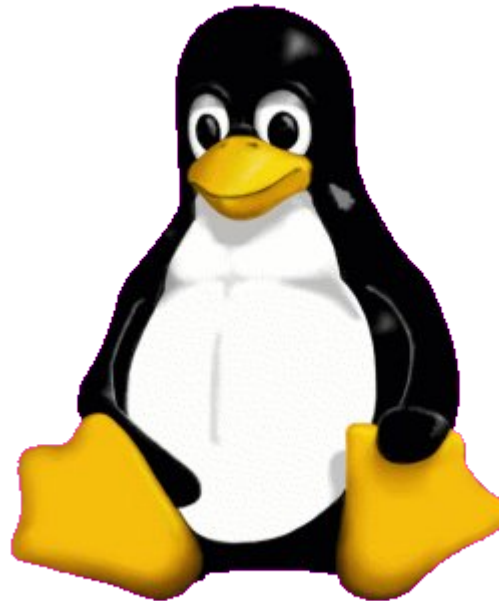
Microsoft's contribution of 22,000 lines of code to the Linux kernel in 2009 wasn't so much an altruistic move (as is commonly portrayed), as it was a move to help Microsoft avoid legal troubles. It was discovered that Microsoft had used GPL code in a closed source Hyper-V network driver, which is in direct violation of the GPL. To remedy the situation, Microsoft contributed the drivers to Linux. It was portrayed by Microsoft as a charitable act, but was more intended to help Microsoft avoid legal entanglements.

In 2009, the 22,000 lines of code contributed to the Linux kernel made Microsoft one of the largest contributors. By 2011, they had fallen to 17th. By 2015, they were no longer even in the top 30 contributors.

The animosity between Microsoft and the Linux community is nothing short of legendary. The tension climaxed between 1998 and 2004 when internal confidential Microsoft documents, known as the [Halloween Documents](#), were released. These documents branded Linux a threat to the Microsoft platform and their business model. You can read them, in their entirety and with annotations, by following the link included above. Even the infamous Steve Ballmer branded Linux a "cancer" and "communist."

Correctly or not (a hotly debated topic), those Halloween Documents assert that Linux's market penetration is greater than that of Apple's Macintosh computers, and that there are more Linux users than Mac users.

In 1996, Linus Torvalds announced that a penguin would be the official mascot of Linux. He decided on a penguin after being bitten on the finger by a small penguin at the National Zoo & Aquarium in Canberra, Australia.



Tux, the Linux mascot, was created by Larry Ewing when, as he states, he should have been studying for finals at college. Ewing is a developer on the Gimp team, and assists with the development of F-Spot, Evolution, and GtkHTML. You can view his story about the creation of Tux [here](#). His Tux page is [here](#). Ewing, who lives in Austin, Texas, created Tux in Gimp 0.54. Neal Tucker created a scalable graphics version of Tux based on the black and white version that was drawn in Gimp. [PC World](#) also ran an article in 2003 about Tux becoming the logo/mascot for Linux.

Over 95% of the world's supercomputers run Linux.

Android, Google's mobile operating system based on Linux, is fast becoming the world's most popular

operating system. Besides Android, the Google Chrome operating system that runs on Chromebooks runs on the Linux kernel.

Chances are high that the menu system in your DVD player, BluRay player, and modern TV are run by an embedded Linux operating system. Linux is also making huge inroads on IoT (Internet of Things) appliances. Furthermore, you're likely to see Linux in your new cars in the very near future, as most major automotive manufacturers have embraced a new, robust "automotive Linux" standard.

Linux forms the backbone of the internet, with over 67% of servers (overall) running Linux, and with over 57% of web sites running on Linux servers.

Thanks to the security built into Linux, there are NO viruses (virii) in the wild for Linux. Occasionally, security vulnerabilities are found (there is no such thing as a 100% secure operating system), but those vulnerabilities are typically patched and closed within just a few days of their discovery. All Linux virus reports are simply "proof of concept" viruses conducted in a laboratory, and typically don't exist in the wild. In the lab, certain criteria must be met in order for the virus to even have a chance to infect a system, and that criteria would almost never exist in the wild (hence the reason to NEVER run your computer as root, except for periodic maintenance tasks).

Distrowatch lists 779 different Linux distros, with 252 of them listed as "active."

As of 2006, Linus Torvalds had written about two percent of the Linux kernel code. Despite the seemingly small percentage, it makes him one of the largest contributors, considering that thousands of people have made contributions. Today, he is the

maintainer of the Linux kernel, merging code into the kernel from other contributors. Torvalds has final say over what gets included and what is omitted.

Linus Torvalds was granted the trademark to the Linux name after several people from different countries attempted to trademark the name in 1994 and 1995. Torvalds subsequently transferred the Linux trademark to Linux International. Today, the trademarked name is protected by the Linux Mark Institute.

Linus Torvalds has become a legend not only for creating Linux, but also for his responses to Linux kernel contributors on the Linux kernel mailing list. In person, he's not much better. Let's just say that Torvalds isn't afraid to say the first thing that pops into his head, and ... well ... simply doesn't have much of a filter. He simply isn't afraid to call it as he sees it. You can read some of his most famous quotes [here](#). Whether you agree or disagree with his tactfulness (or lack thereof), it certainly keeps things interesting.

Since its inception on the Intel 386 processor in 1991, Linux has been ported to most every other computer platform.

The [corporate membership](#) of the Linux Foundation reads like a who's who of leading corporations from across many walks and disciplines. They include IBM, NEC, Google, Adobe, Mazda, Hitachi, Samsung, Toshiba, Toyota, AMD, Intel, Bosch, Denso, Honda, Epson, Cisco, HP, Fujitsu and many, many others.

The Linux kernel is released under the GNU GPLv2 license. Torvalds and many of the other kernel developers decided against adopting the newer GNU GPLv3 license when it was released in 2007.



PCLOS-Talk

Instant Messaging Server

Sign up TODAY! <http://pclostalk.pclosusers.com>



Screenshot Showcase



Posted by martin, July 10, 2016, running e20.

ms_meme's Nook: Today While I Boot Up

Today while I boot up to see my desktop
Without going further
I pause to give thought
For a million boot ups PCLOS will stay
I'll not forget all the joy
It gives me every day

I am content with all of its glories
Ease and repose
To me it does bring
I know of its praises from user stories
And that is why I do sing

I think of the forum and friends I have met there
All different ages
And from everywhere
The sandbox always is in full swing
Fun to you they bring

It is a dandy I won't be a rover
Know that I love it
By songs that I sing
We all flock to it like bees to clover
Texstar of this OS is king

MP3

OGG

PCLinuxOS Recipe Corner



*from the kitchen of
youcantoo*

Slow-Cooker Sloppy Joe Lasagna

Ingredients

1 pound ground beef
1/2 large white onion, diced
2 carrots, diced
1 red bell pepper, diced
1 green bell pepper, diced
3 cloves garlic, minced

1 (6 ounce) can organic tomato paste
2 tablespoons ketchup
2 tablespoons Worcestershire sauce
2 (14.5 ounce) cans fire roasted diced tomatoes
1 pinch of salt and pepper
6 to 8 wide and flat no-boil lasagna noodles
2-3 cups shredded cheddar cheese



Directions

1. In a large sauté pan, brown the beef over medium-high. If using a high-fat beef, drain. Remove meat from skillet; set meat aside and return pan to the stove.

2. Back in the pan, add the onions and carrots. Sauté for five minutes, until they start to soften. Add the peppers and garlic. Sauté another 2 to 3 minutes, taking care to not burn the garlic.

3. Add the tomato paste, ketchup and Worcestershire sauce. Stir to combine. Add the beef and tomatoes to the pan and simmer for a few minutes. Taste it and season with salt and pepper if needed.

4. In a slow cooker, spoon a little of the beef mixture on the bottom. Lay a couple of lasagna noodles on top of the beef, breaking them to fit the pan. Layer with beef mixture, more noodles, more beef, more noodles and the remaining beef. Sprinkle the cheese on top.

5. Cook at medium for 3 to 4 hours, until the noodles are cooked and cheese is melted.

Serves 6



**The
PCLinuxOS
Magazine**

**Created with
Scribus**

Inkscape Tutorial: A Painted Text Effect

by Meemaw

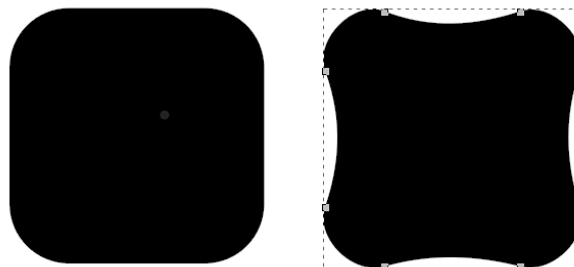
I found a tutorial which shows how to create an image that looks like it was painted on some sort of material.



We used to see burlap bags of materials with the labels painted on them. We should do something like that.

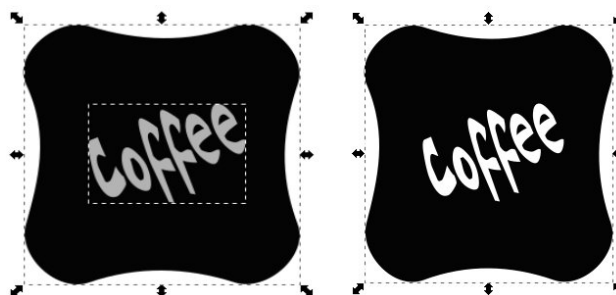
You will need two bitmaps of the same texture (one normal color and one greyscale) to use for the bag. I am using a burlap pattern I got from www.textures.com, but you can get them [here](#).

We will make the label. Open Inkscape and draw a rectangle or some shape you think would make a great label. Make sure you change it to a path using Path > Object to Path. I made a 400 x 400 px square with rounded corners, then curved the sides inward (center, top).



Now create your text, and change it to a path. Depending on your version of Inkscape, this involves choosing **Path > Object to path**, then **Ungrouping** the text, then choosing **Path > Union**. I made my text grey, so I could see it against the black label (below, left).

Place your text over the label, choose both and do **Path > Difference**. Now it looks like the text has been cut out.



Now we will make the painted layer. With the greyscale texture, position the label behind it and choose **Object > Mask > Set** (top, right).



Select and align the greyscale with the color texture so it looks like the textures match. If the label layer isn't dark enough, just duplicate it.



Save your work and export your creation.

Extending The Life Of 32 Bit Chrome On PCLinuxOS

by Alessandro Ebersol (Agent Smith)



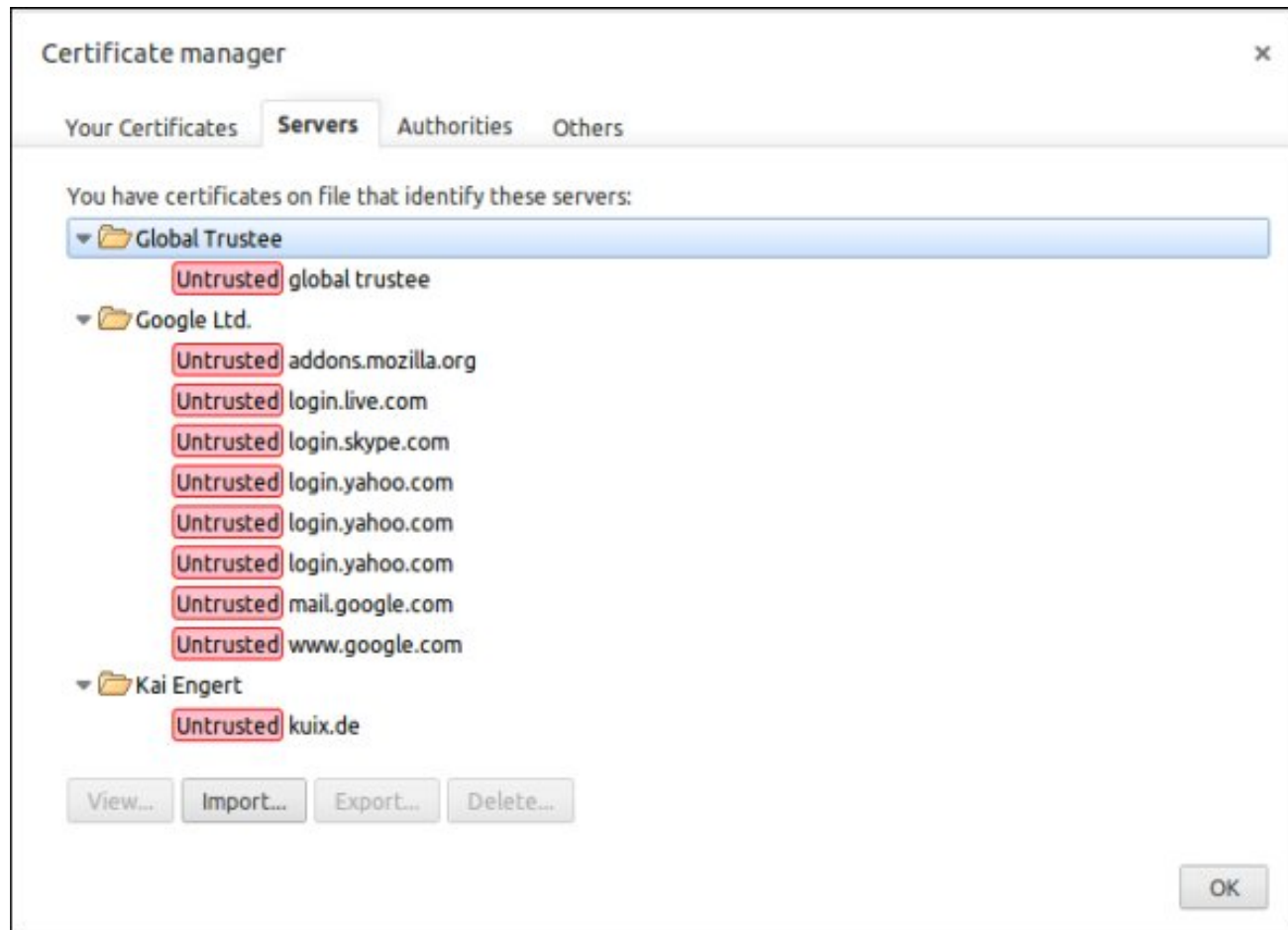
Google ended its support for Chrome on Linux 32-bit. If you try to use it, you can no longer access the services from Google itself (YouTube, Gmail, Google search, etc ...), because SSL certificates are no longer valid. You will have a screen like the one above.

Interestingly, you can access any other site and it is Google's deliberate choice to end Chrome for 32 bit Linux through this kill switch, after all. Today, Google is so inserted in the internet infrastructure that is difficult to avoid it.

Just to illustrate, try to access the settings of your Chrome for Linux 32-bit, and go to the Advanced section, HTTP/SSL, manage certificates. You should see something like the picture above and to the right:

Clicking on any of the Google domains will invariably appear as expired, which explains the refusal of access to any of the Google services. However, you can extend the life of Chrome on Linux 32-bit, either to watch Netflix or play a Facebook game in Flash.

How? There is an option to change the security protocol, to TLS 1, in order to make Chrome access sites with SSL certificates. Therefore, it is necessary



to invoke Google Chrome with “--ssl-version-max=tls1”.

So, the command would be:

```
/usr/bin/google-chrome-stable %U --ssl-version-max=tls1
```

By doing so, Google Chrome will return to work, even with the sites with expired SSL certificates.

To automate the process, and end the problems with Chrome, you need to change your .desktop file of Google Chrome. The .desktop file is located in /usr/share/applications, and is called google-chrome.desktop. Edit it with a plain text editor, as root, and find all lines with:

Exec=/usr/bin/google-chrome-stable %U

Replace it with:

Exec=/usr/bin/google-chrome-stable %U --ssl-version-max=tls1

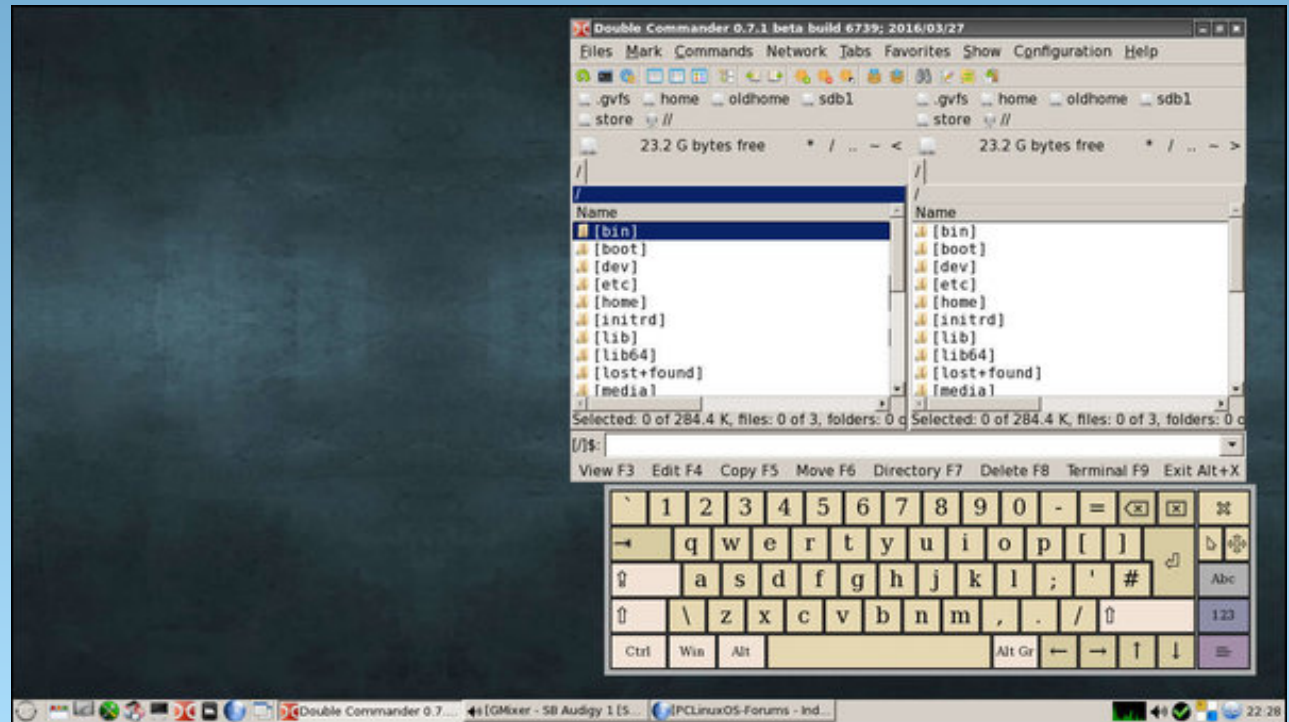
Save the file, and watch Google Chrome start to work again.

Note: Take care when using an older version of Google Chrome, as security vulnerabilities will not be patched.

Have fun with your Google Chrome on Linux 32-bit.



Screenshot Showcase



Posted by amlu, July 6, 2016, running LXDE.



Does your computer run slow?

Are you tired of all the "Blue Screens of Death" computer crashes?



Are viruses, adware, malware & spyware slowing you down?

Get your PC back to good health TODAY!

Get



Download your copy today! FREE!

Tip Top Tips: Kodi With The Latest Add-ons

Editor's Note: Tip Top Tips is a new monthly column in The PCLinuxOS Magazine. Each month, we will feature – and possibly even expand upon – one tip from the PCLinuxOS forum. The magazine will not accept independent tip submissions specifically intended for inclusion in the Tip Top Tips column. Rather, if you have a tip, share it in the PCLinuxOS forum's "Tips & Tricks" section. Your tip just may be selected for publication in The PCLinuxOS Magazine.

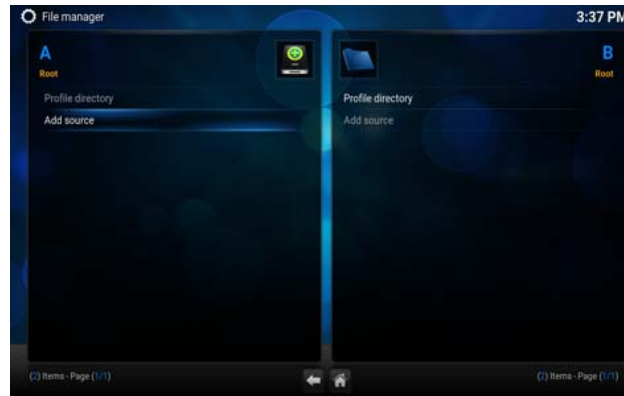
This month's [tip](#) comes from PCLinuxOS forum member [trytip](#).

It's best to start with a FRESH Kodi, so DELETE or RENAME your old .kodi. This addon will install the latest EXODUS, which used to be GENESIS, plus a lot MORE. It will customize the background and ADD a few shortcuts to the HOMEPAGE. This is by FAR the best STARTER package which includes VIDEO/MUSIC/PROGRAMS.

Step 1: Navigate to the "File manager" tab under the "SYSTEM" menu heading.



Step 2: Click on the "Add source" function at the left side of the screen.



Step 3: Click on the top text input box marked "<None>" under the "Enter the paths or browse for the media locations" field.



Step 4: Input "http://fusion.tvaddons.ag" (without quotations) then Press the "Done" button.

<http://fusion.tvaddons.ag>



Step 5: Click on the bottom text input box under the "Enter a media name for this media source" field.



Step 6: Input the word "fusion" (without quotations) then Press the "Done" button.

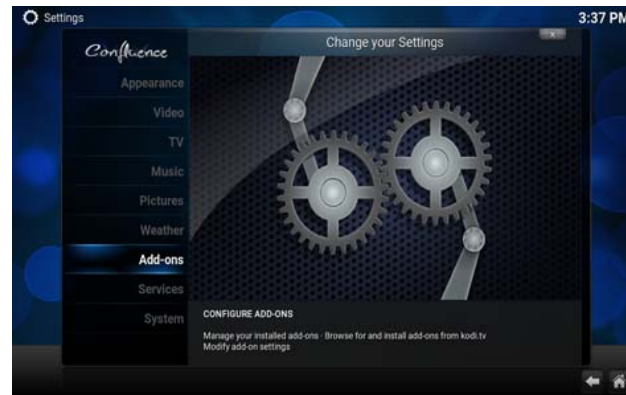


Part 2 – Run Config Wizard

Step 1: Return to the Kodi home screen, then Navigate to the “Settings” tab under the “SYSTEM” menu heading.



Step 2: Select the “Add-ons” tab from the left side vertical menu.



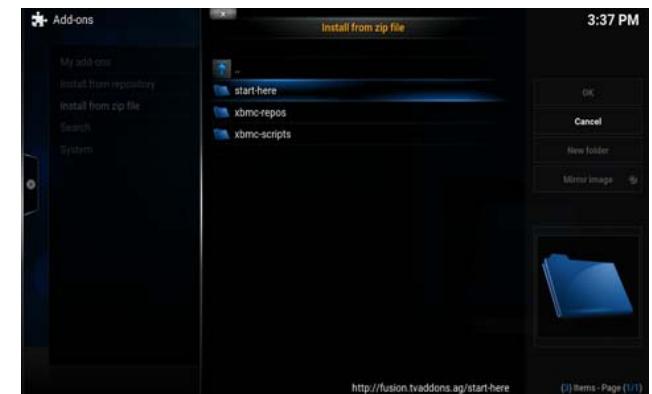
Step 3: Choose the “Install from zip file” function from the listing.



Step 4: Click on the “fusion” server folder at this point.



Step 5: Choose the “start-here” folder from the listing.



Step 7: Click on the “OK” button at the bottom of the window to save the file source at this point.



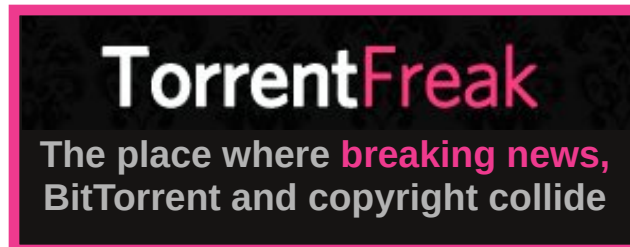
Step 6: Click on the “plugin.video.hubwizard-1.1.9.zip” file at this point.



Step 7: Return to the Kodi home screen, then Navigate to the “PROGRAMS” menu.



Step 8: Launch the “Config Wizard” tool from the listing.



Step 9: Choose your desired operating system (LINUX), they’re basically all the same.



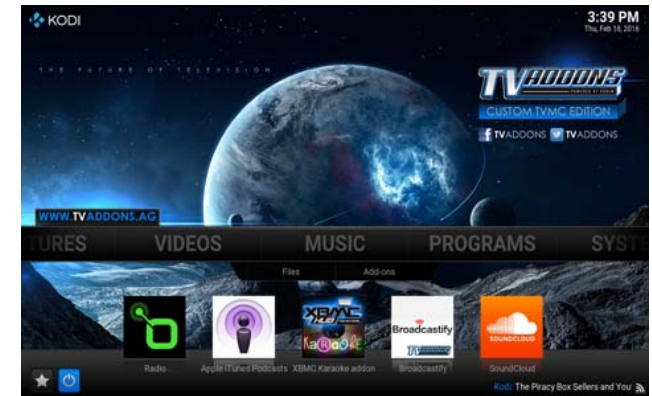
Step 10: Press the “Yes” button when prompted to confirm the customization process.



Step 11: You’ll now have to wait about a minute for the process to complete.

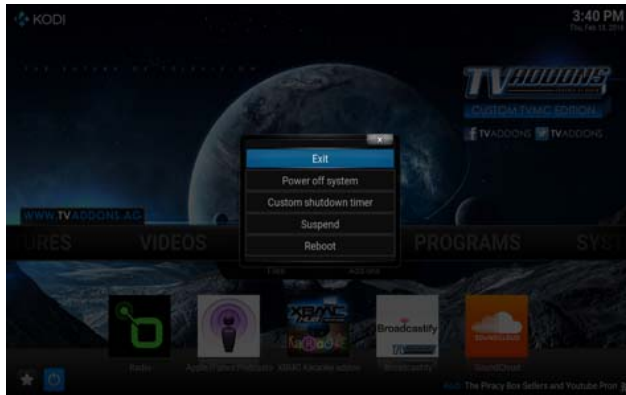


Step 12: Once completed, you’ll be automatically sent back to the Kodi home screen. You’ll now want to click on the small power button located at the lower left corner of your screen.



Step 13: Press on the “Exit” button at this point to close Kodi.





Step 14: You'll now want to relaunch Kodi to find your shiny new configuration ready to use!



KODI gets annoying because nothing is permanent and you constantly need to know what addons are valid, but then again, it's the same with all tv/movie websites and programs like Popcorn Time or Stremio.

You can also use the same REPO and install some OTHER HIDDEN add-ons, like **Erotik (XXX)** or **Genesis** (i guess it's updated). It should already be in your **PROGRAMS/ADD-ONS/ADDON INSTALLER** there are many others, but try one at a time. I don't know them all, since I'm new fairly new to Kodi.

SETTINGS

Sometimes audio is low, but you can amplify it.

Change Wallpaper

Copy any wallpaper to `.kodi/media/`. Then, in **SETTINGS/APPEARANCE/SKIN**, click **settings/background/background path**.

A very useful option is to have Kodi WINDOWED. You can then resize it how you want by dragging the sides. This way, if Kodi freezes, you can quickly KILL with a task manager.

System/Settings/System/Video output/Display mode, then click the resolution up or down till it says **Windowed**.

Subtitles

System/Settings/Video/Subtitles/ scroll down to Default TV show service. Click it and GET MORE, then select **OpenSubtitles.org**. Then, do it again and add **Subscene.com**. Now you can use the subtitle icon. **WARNING:** go to `.kodi/media/` and create a folder called Subtitles. Then, in **System/Settings/Video/Subtitles/** Subtitle storage location, click **Custom location**, then underneath select Custom location folder.

TO VIEW HIDDEN FOLDERS
System/Settings/Appearance/File lists/Show hidden files and directories.



Support PCLinuxOS! Get Your Official

PCLinuxOS
Merchandise Today!

PCLinuxOS

Repo Review: Astronomy Programs

by Cg Boy

Astronomy might not be for everyone. For some, astronomy is just looking up at a lot of boring stars. But for others, astronomy is looking out into space, at the wonder, the beauty, of the universe. In this short article, I'll review some astronomy related applications you can find in the PCLinuxOS repository.

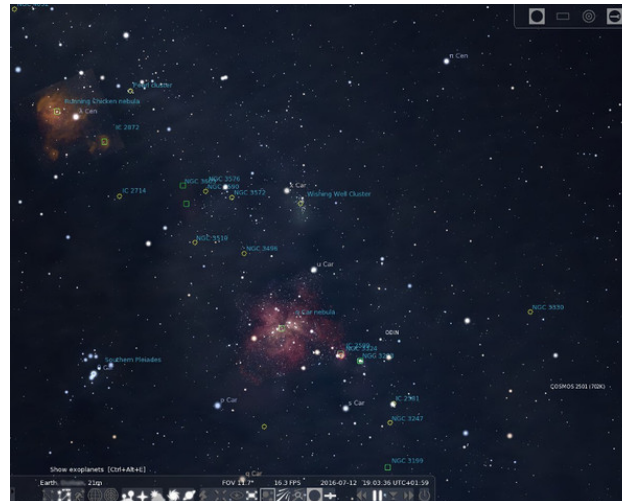
Astronomers used to use star charts to locate objects they wanted to observe. Of course, a lot still use them. But Planetarium software can simulate the time of day and year, location on Earth, locations of stars, nebulae, galaxies, star clusters, planets, comets, asteroids, and with some software, satellites.

Let's have a look at what there is in the repository.

Stellarium. As much as I like KStars, I'm giving Stellarium first place, mostly because of its ease of use and good graphics. Stellarium is pretty user

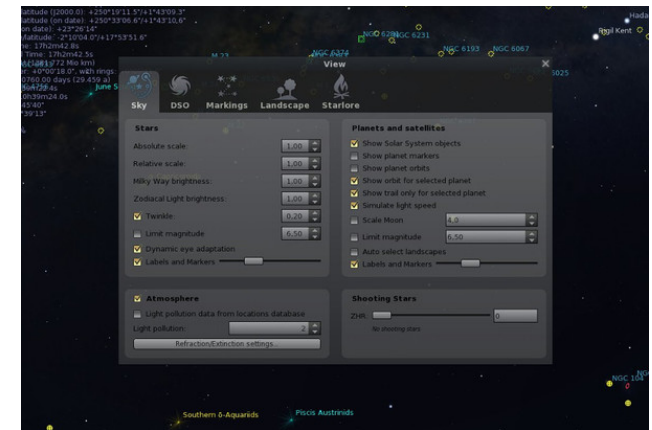
friendly, therefore making it fairly easy to locate objects. Stellarium also has a lot of imagery for galaxies, nebulae, star clusters, etc. It has the ability to show the locations of satellites, though I'm not sure how accurate it is. It comes with catalogs of comets and asteroids, but you'll probably want to update that data as new comets and asteroids are discovered. You can also download more star catalogs.

Stellarium can control GoTo telescopes, but since I don't own one, I can't test that feature out. Stellarium can simulate equatorial mount movement. It comes with a variety of interesting plugins, like "Meteor Showers," "Exoplanets," "Quasars," "Bright Novae," "Observability Analysis," "Oculars," and "3d Sceneries." "Observability Analysis" will display rise and set info for the selected object. With the "Oculars" plugin, you can get an idea of what the view through a telescope or camera might be like.



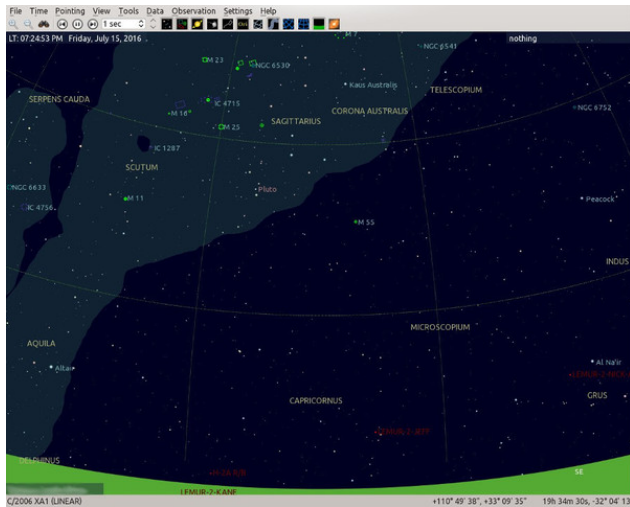
In the "Sky and viewing options window," you can set the light pollution level, which could be useful. Stellarium can simulate star twinkling and also

dynamic eye adaption. The latter meaning that when there's a bright object, like the Moon, near a faint object, the faint object will dim even more, if not become invisible. There are also a number of different landscapes you can choose from. You can even load in more landscapes.

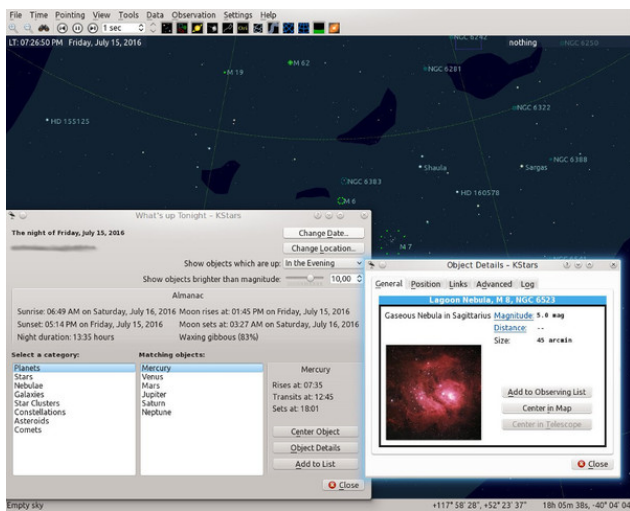


All in all, Stellarium is an excellent program. It's fairly easy to use, and the 3D graphics are really great.

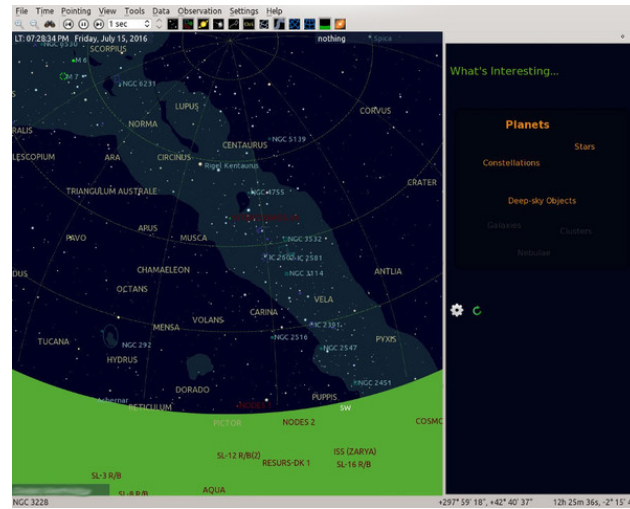
KStars. Stellarium is great and all, but KStars is probably still my favorite of the three. Why? Because, although it doesn't have the fancy graphics, it has useful tools and features that Stellarium just doesn't have. The "What's up Tonight" tool is quite useful. It shows you what will be in the sky tonight. And there's the "What's Interesting..." tool, which shows you what's currently in the sky. It has other interesting tools and features, but one useful tool is the "Observation Planner". You can use it to make a list of objects you wish to observe. Then, if you click "Execute the session Plan," you can log the observations you make. Some of the other tools include: "Sky Calendar," "Moon Phase Calendar," "Altitude vs. Time," "Solar System," and "Jupiter's Moons."



KStars also has satellite tracking, but again, I don't know how accurate it is. Like I said, KStars doesn't come with images of galaxies, nebulae, etc. But you can download add-ons for this. Unfortunately, the add-on only provides images for the Messier catalog of objects. You can also right-click on an object and click "Show SDSS Image" or "Show DSS Image," which will download an image of that area of sky. You can also download more catalogs of stars and other objects.



KStars is supposed to be able to control telescopes, but the version in the repository doesn't appear to have that ability. It might be that the package in the repository wasn't built with the INDI library. I'm not sure. KStars is fairly user friendly and easy to use, but still not as much as Stellarium.



Anyway, KStars is awesome, and although it doesn't have the fancy graphics of Stellarium, it still has some great features. Unfortunately, I think the version in the repository is out of date.

SkyChart/Cartes du Ciel. And now we get to the third, and most complicated one. I wouldn't really call Cartes du Ciel easy to use. I think it might be better suited for more advanced astronomers, not really beginners. Navigating is still pretty similar, though you use the middle mouse button instead. The interface is a bit cluttered, though. But it is a powerful program (top, right).

It can control motorized telescopes (can't test that out because I don't have one). Unfortunately, like KStars, it doesn't come with images of galaxies, nebulae, etc. But like KStars, it has the ability to get a DSS image of an object. You can get the image by clicking on the little button that says "DSS." You'll have to zoom in quite a lot to be able to get the DSS

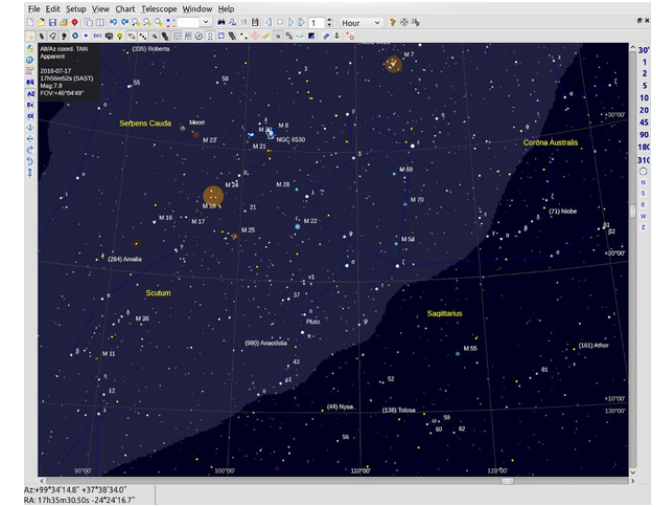
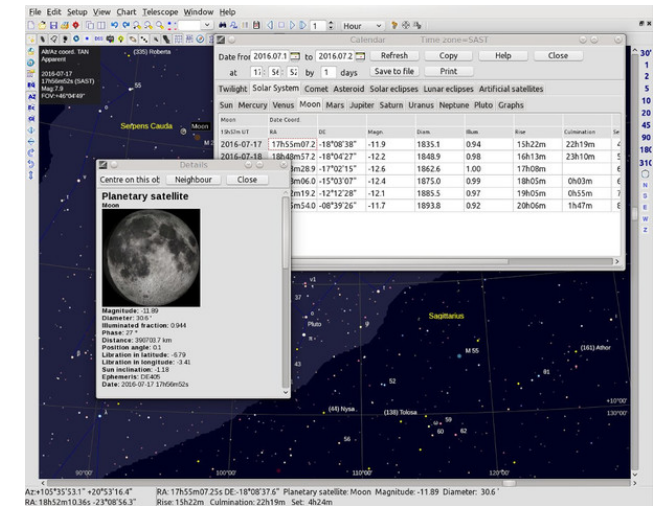
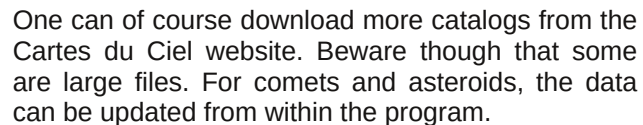


image. You can also download and install an image package from the website. I did not do this, as the package is over 100mb to download, and I have a bandwidth limited internet connection.



One useful tool is the "Ephemeris calendar". It shows you information for twilight time, the Sun, the Moon, the planets, comets, asteroids, solar and lunar eclipses, and artificial satellites. It can show information like location, magnitude, diameter, rise and set times, dates, and other info. Now, it's

Another tool is the “Object list,” which list objects in the current field of view. You can configure what kinds of objects it will show. And there’s also “Solar System Informations”. It shows a planet visibility graph, what the planets and the Moon currently look like, and a diagram of the solar system.



This is a great program, but I personally prefer KStars and Stellarium. To me Cartes du Ciel is just too slow, and too hard to use. Unfortunately, the version in the repository is out of date.

PCLinuxOS Magazine



So, my advice would be, if you are beginning astronomy, use Stellarium. When you get more experienced, use KStars. When you're even more experienced, use Cartes du Ciel. But that's just my advice.



Support PCLinuxOS! Get Your Official
PCLinuxOS
 Merchandise Today!

PCLinuxOS

Disclaimer

1. All the contents of The PCLinuxOS Magazine are only for general information and/or use. Such contents do not constitute advice and should not be relied upon in making (or refraining from making) any decision. Any specific advice or replies to queries in any part of the magazine is/are the person opinion of such experts/consultants/persons and are not subscribed to by The PCLinuxOS Magazine.
2. The information in The PCLinuxOS Magazine is provided on an "AS IS" basis, and all warranties, expressed or implied of any kind, regarding any matter pertaining to any information, advice or replies are disclaimed and excluded.
3. The PCLinuxOS Magazine and its associates shall not be liable, at any time, for damages (including, but not limited to, without limitation, damages of any kind) arising in contract, tort or otherwise, from the use of or inability to use the magazine, or any of its contents, or from any action taken (or refrained from being taken) as a result of using the magazine or any such contents or for any failure of performance, error, omission, interruption, deletion, defect, delay in operation or transmission, computer virus, communications line failure, theft or destruction or unauthorized access to, alteration of, or use of information contained on the magazine.
4. No representations, warranties or guarantees whatsoever are made as to the accuracy, adequacy, reliability, completeness, suitability, or applicability of the information to a particular situation. All trademarks are the property of their respective owners.
5. Certain links on the magazine lead to resources located on servers maintained by third parties over whom The PCLinuxOS Magazine has no control or connection, business or otherwise. These sites are external to The PCLinuxOS Magazine and by visiting these, you are doing so of your own accord and assume all responsibility and liability for such action.

Material Submitted by Users

A majority of sections in the magazine contain materials submitted by users. The PCLinuxOS Magazine accepts no responsibility for the content, accuracy, conformity to applicable laws of such material.

Entire Agreement

These terms constitute the entire agreement between the parties with respect to the subject matter hereof and supersedes and replaces all prior or contemporaneous understandings or agreements, written or oral, regarding such subject matter.

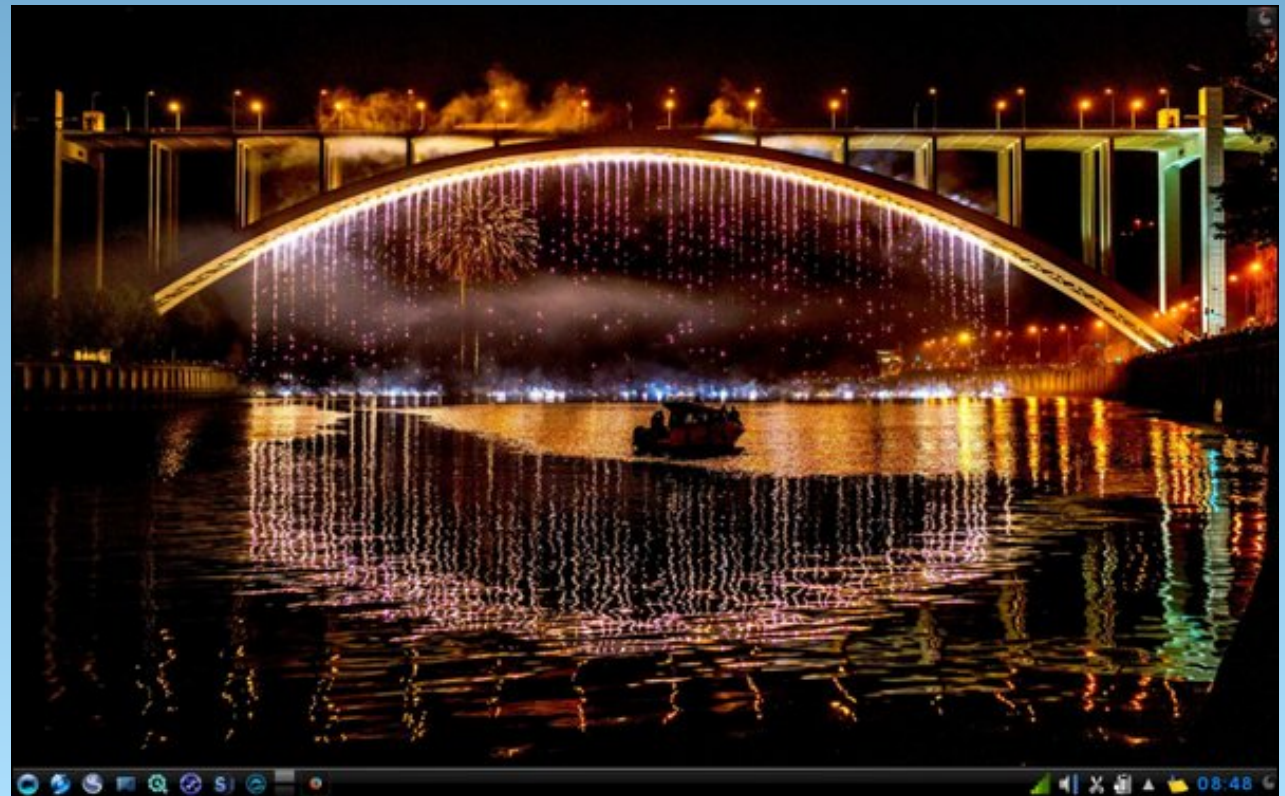


CHIMPBOX

The chimpbox packs a punch. Zero noise, small footprint and low power usage.

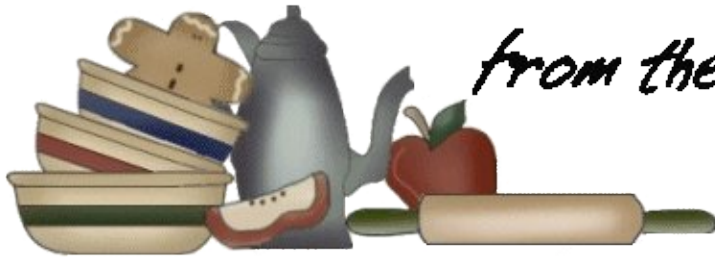
<http://chimpbox.us>

Screenshot Showcase



Posted by luikki, July 4, 2016, running KDE.

PCLinuxOS Bonus Recipe



*from the kitchen of
youcantoo*

Watermelon Fire and Ice Salsa

A summertime treat with an unbelievable blend of flavor and textures.

Ingredients

3 cups cubed watermelon
1/2 cup chopped green bell pepper
2 tablespoons lime juice
2 tablespoons chopped fresh cilantro
1 tablespoon chopped green onions
1 tablespoon chopped jalapeno pepper
1/2 teaspoon garlic salt

Directions

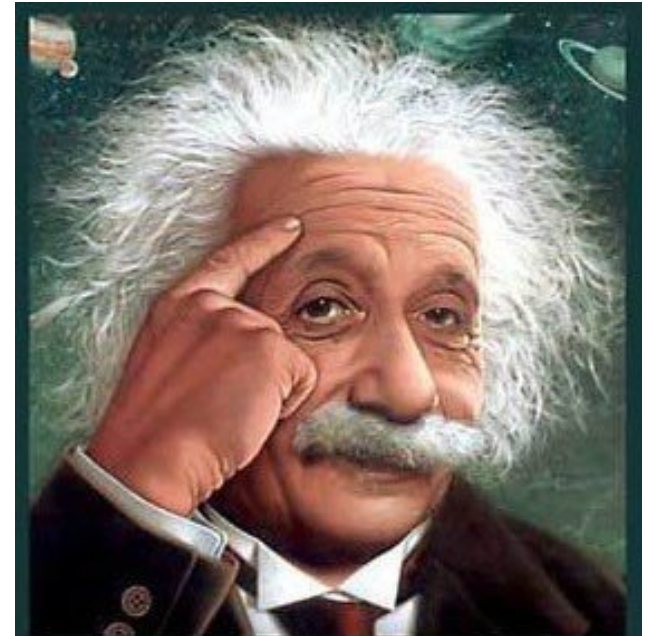
In a large bowl, combine the watermelon, green bell pepper, lime juice, cilantro, green onions, jalapeno and garlic salt. Mix well and serve.

Options: Serve with Lime Tortilla chips.



The
PCLinuxOS
Magazine

Created with
Scribus



It's easier than $E=mc^2$
It's elemental
It's light years ahead
It's a wise choice
It's Radically Simple
It's ...



PCLinuxOS Family Member Spotlight: craesz

As told to YouCanToo



What is your name/username?

craesz

How old are you?

I'm just north of 50.

Are you married, single?

Divorced, single, but severely involved with one of the most beautiful redheads on the planet!

How about Kids, Grandkids ?

I have two children. My son will be 25 in August. He's pretty amazing. Because of my divorce, I haven't had much input in certain areas, and his college education was not up to me. I was extremely skeptical about his choice [in the arts]. He has since made me eat a little crow [no offense Crow] because he obtained in his first job, not only a job in his field, but he has excelled. Apparently they had an industry gathering recently, and he was the only one from his company to walk away with a gold medal. I'm pretty sure those genes came from someone else. My daughter was adopted from Guatemala. She will be 17 in November. She is a gem. The Guatemalans are an amazing group of people that sincerely love their children. I am so happy to have her.

Do you have pets, what is your favorite?

Pets..... swore off those when I had a most amazing

cat that I had to take to the vet for... well... you know... I cried like a baby. Not doing that again.

My girlfriend's daughter has a Yorkie-Poo that has taken a severe liking to me. It's not that I don't like animals... just can't take that later stuff.

Are you retired, still working and if working, what do you do?

Still working. Will have to do so until I die. I have a degree in Computer Science [probably why I love Linux] and programmed for I.B.M. while I was in school. I want to help more with PCLinuxOS, but my time is stolen almost 24/7 between work and home. I have a current plumbing issue now that just won't go away. I have to upgrade suspension on my car. If I could quit it all and just help with PCLinuxOS, that would be amazing, but doubt that would pay the bills. Unfortunately, got a little child support to pay and those bills just keep coming.

Where do you call home? What is it like? IE: weather, scenery

I am north of the Dallas/Fort Worth area. It's hot in the summer and in winter we typically get the worst type of winter weather. For some reason, we are in the perfect place for ice. Not snow, ICE. When it comes, none of these idiots know how to drive on it. Of course, none of the cities are properly prepared.

Where did you go to school and what is your education level?

All school was in Texas. Post-graduate.

What kind of things you like doing? hobbies, travel, fishing, camping?

Hobbies, not really. Have to deal with work, take care of a couple of kids, two properties next, helping family [extended and otherwise], then whatever is left is for fun. Golf every so often. Really like to cook.

Unfortunately, I have turned into THE cook. Ends up that I cook when I'm not doing any of the other stuff. Somehow that cooking talent comes naturally to me, so everyone else refuses once they figure out I can. Good thing I like it. I would say that I watch the "food" channels a lot. Some of that I have implemented and it worked out. Pretty cool.

Why and when did you start using Linux?

This is a long story. I tried to upgrade a Win95 machine. I found out the hard way that if you change the hardware, Windows would lock you out. I was just a little upset about it. (Okay, that is a severe understatement, EXTREMELY upset!!) I started looking into Linux and PCLinuxOS was the only system that recognized, without difficulty, my hardware, etc. So, given my background, I decided to kick M\$ to the curb and move entirely to Linux. If it weren't for the ease of PCLinuxOS, I would not have done this. I recently got rid of every M\$ machine in my office. The only issue left is how the rest of the world works. I do think it is funny when I tell someone "I don't use Windows" and they have this glassy eyed look.

I am so very glad Tex exists. My life would be hell without him.

PCLinuxOS Family Member Spotlight is an exclusive, monthly column by YouCanToo, featuring PCLinuxOS forum member. This column will allow "the rest of us" to get to know our forum family members better, and will give those featured an opportunity to share their PCLinuxOS story with the rest of the world.

If you would like to be featured in PCLinuxOS Family Member Spotlight, please send a private message to youcantoo, parnote or Meemaw in the PCLinuxOS forum expressing your interest.

Game Zone: Empire Of The Gods

by daiashi

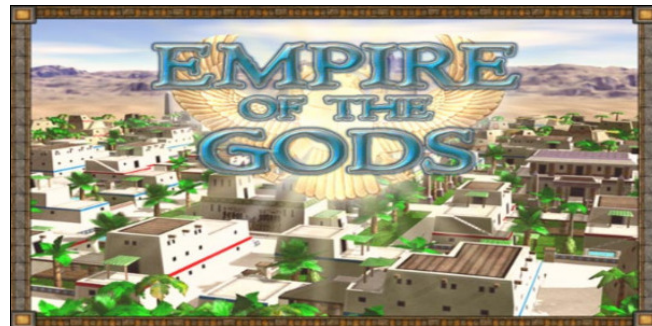
almost seems non-ending at times. If you have some time, try out the demo. Build your ancient city.

System requirements:

Fully updated PCLinuxOS and Steam

Hardware:

OS: PCLinuxOS
Processor: 1.0 GHz
Memory: 1 GB RAM
Graphics: 128 MB, OpenGL 2.0
Storage: 95 MB available space



About The Game

Build flourishing cities and the largest monuments ever seen.

Enjoy an original card game with absorbing gameplay which whisks you back several centuries to the times when people built huge, unbelievable monuments. Work your way up from a small tribe to a flourishing Empire along the Nile. Build the large pyramids which tower above vast cities, win the Gods' favour and become the mightiest Pharaoh. The Gods are known to favor the mightiest one. Can you architect a new, unstoppable Empire of the Gods?

- Enjoy an original and simple card game.
- Build 12 beautiful cities and monuments.
- Improve your character.
- Use dozens of bonus and extra cards.
- Reach up to 15 achievements.

I enjoy it. You may not. It's a different kind of game. No exploding, guns or fire. There is some sort of learning curve as to what cards to choose and not to choose for your task on hand. It is a long game. It

About The Company

They are an independent game development team from Slovakia. They have a small two-man team and **Lonely Troops** was founded in 2004 by brothers Jaroslav and Tomáš Kurčík. Their goal is to make fun and original computer games for casual players. They have done just that with this title.

Some Gameplay Screenshots





PCLinuxOS All your
Connect connections in one
convenient location!



Getting It To Run

Install Steam (if you don't have it installed already), then start it. You will need to create a new account, if you do not already have one. Once you have Steam up and running, go to the store tab. Click on the Linux tab if you wish and search for Empire of the Gods. Click on and download the demo. If you have updated your system, including graphics drivers, you should be good to go.

Empire Of The Gods



Chat with PCLinuxOS users
from all over
the World.

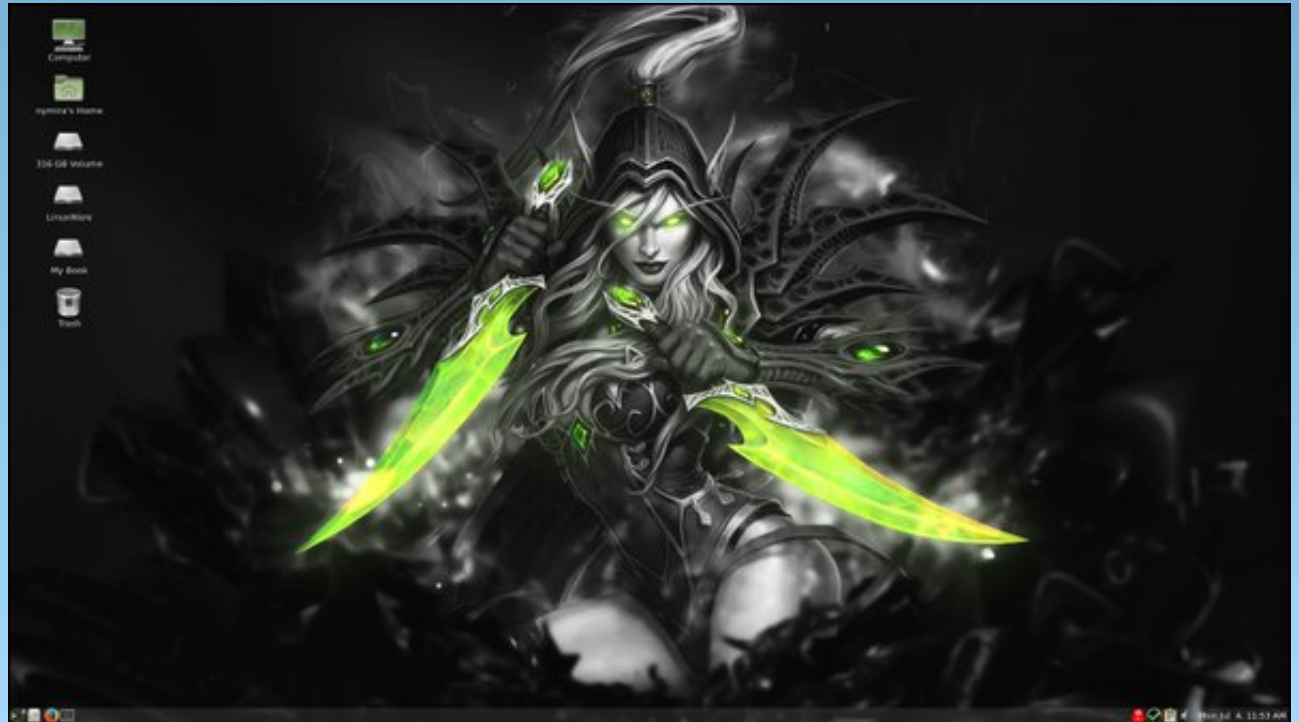


**Sign up
TODAY!**

PCLOS-Talk

<http://pclostalk.pclosusers.com>

Screenshot Showcase



Posted by nymira, July 4, 2016, running Mate.

PCLinuxOS Puzzled Partitions

	3		2		4			8
4			8				9	
	9					5	7	
2	4				9			
						3	2	1
6				3				
				4				
					7		8	
7	5		3					9

SUDOKU RULES: There is only one valid solution to each Sudoku puzzle. The only way the puzzle can be considered solved correctly is when all 81 boxes contain numbers and the other Sudoku rules have been followed.

When you start a game of Sudoku, some blocks will be prefilled for you. You cannot change these numbers in the course of the game.

Each column must contain all of the numbers 1 through 9 and no two numbers in the same column of a Sudoku puzzle can be the same. Each row must contain all of the numbers 1 through 9 and no two numbers in the same row of a Sudoku puzzle can be the same.

Each block must contain all of the numbers 1 through 9 and no two numbers in the same block of a Sudoku puzzle can be the same.



SCRAPPLER RULES:

1. Follow the rules of Scrabble®. You can view them [here](#). You have seven (7) letter tiles with which to make as long of a word as you possibly can. Words are based on the English language. Non-English language words are NOT allowed.

2. Red letters are scored double points. Green letters are scored triple points.

3. Add up the score of all the letters that you used. Unused letters are not scored. For red or green letters, apply the multiplier when tallying up your score. Next, apply any additional scoring multipliers, such as double or triple word score.

4. An additional 50 points is added for using all seven (7) of your tiles in a set to make your word. You will not necessarily be able to use all seven (7) of the letters in your set to form a "legal" word.

5. In case you are having difficulty seeing the point value on the letter tiles, here is a list of how they are scored:

0 points: 2 blank tiles

1 point: E, A, I, O, N, R, T, L, S, U

2 points: D, G

3 points: B, C, M, P

4 points: F, H, V, W, Y

5 points: K

8 points: J, X

10 points: Q, Z

6. Optionally, a time limit of 60 minutes should apply to the game, averaging to 12 minutes per letter tile set.

7. Have fun! It's only a game!

E₁ G₂ L₁ J₈ V₄ U₁ G₂

Blank tiles

Triple Word

P₃ C₃ N₁ O₁ O₁ R₁ P₃

Blank tiles

C₃ N₁ L₁ Z₁₀ A₁ E₁ D₂

Blank tiles

P₃ E₁ M₃ R₁ O₁ F₄ R₁

Blank tiles

H₄ G₂ O₁ R₁ P₃ E₁ N₁

Blank tiles

Double Word

Possible score 279, average score 195.

Download Puzzle Solutions Here

PCLinuxOS Crossword Puzzle: August 2016

Carnival

```

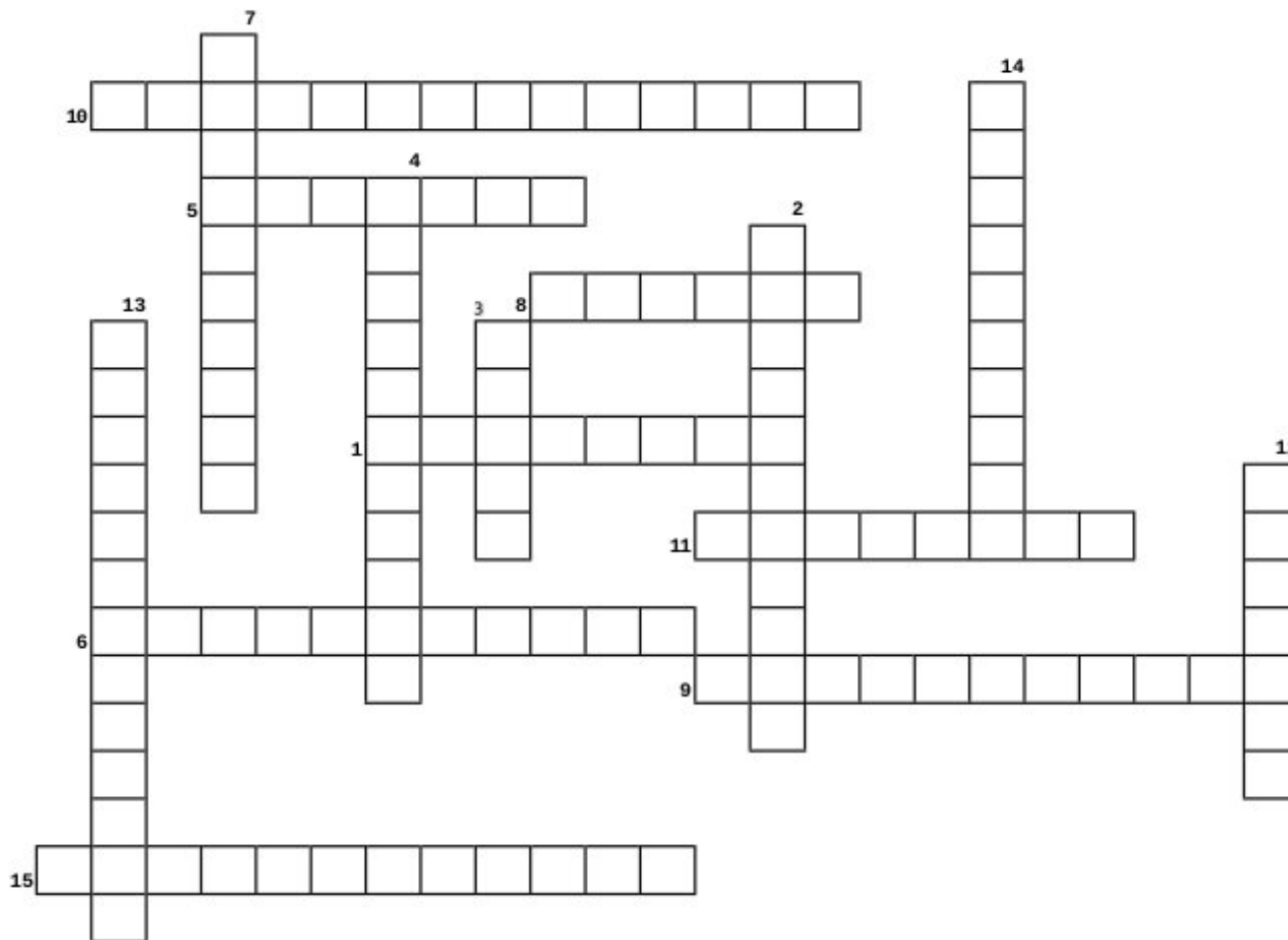
V M A C K E I L G Y Z R J M V A I D H Y B Q T Y L F G D Q K
D O U W T W S I A D B Q N W F H G B V N H Q F N B Q W N G G
M M B H T R L L N H E B H V R V J U J G U G E E Z G G X T N
L Z J B K F A L M D N S K A X K J X U A A C J F W U C E T X
T R M V Y F N N O H F W U G O D T O H G Z E A E L W A N Z S
B G A G A I F T I L U P H O J E D E N J J U A R I D S T D N
A J R C A N D Y A P P L E M H D I Q R A W S W R N H V E U O
H S C S P N G O A J S A L E E N O E E W E Q O I T I B R R I
O Y Y C A X Y Y D M N A E I A K U M N W O L C S Q Y V T K S
M R X V R C P F S N H E R V C X G F T X R D G W M T M A S S
T R P N A S X A K O H F H E E E B C E K G A P H Z C A I L E
Z N P W D K O B V I E K Z B O O C S Q U H O M E V Q I N Q C
X X E O E X E Z S T P V A D S B F R S S K L P E Q B B M P N
D N R F J V M L R I F B A C I E U E E W H I O L D B X E D O
C A F O J V C C L B S R Y P A Q N L O A P V C R Y G M N U C
F O O F O Q R Q P I U K S D M C N G Q L M S V I T I U T J C
K I R W Q W Y T S H B Y S Y N F E G A G F N L K F O R K E L
P L M N H E N B R X E W J C K A L U K X V A Q P R U P C V G
L D A U D G C G O E W Y M L E B C J L C O W A O T N Y I B E
P A N K W O J Z R V U T U S J N A N U N B R G Y E U K R M W
T U C R D L G Y R V O Z S M R O K W O V Z Y B L Y G P C U B
P D E H Q V L T I N X I I O T Y E S E T R Y C R R P G U R U
S R S W A R R D M M L M C J H A S Y A R T S Z J B N A S T M
W B N D Y P E S F M W P K V X J Y U E M I O L E U Q O D I P
Q A O S B H S E O C O R I L H T K M V E K D C T N Q N D E E
T F W G B L X F E P A C F Z C U L L D W N R E T X E W D X R
W G C M Q R K L S A T N E M E S U M A L B C S S P A K H M C
Q V O A J R X N U Q S Y R I S M U E J O S V A S Y O R C Q A
S Q N O A D A E O O N R O M V R O L L E R C O A S T E R Z R
U I E T F R N B H M G V C F B U J H Y C Y Y X L H D E I F S

```

amusement
 bumper cars
 candy apple
 carnival
 circus
 clown
 concessions
 corn dog
 cotton candy
 entertainment
 exhibition
 ferris wheel
 fried dough
 fun house
 funnel cakes
 hot dog
 house of mirrors
 ice cream
 juggler
 mask
 merry-go-round
 midway
 music
 parade
 performances
 popcorn
 rides
 roller coaster
 snow cone
 spin art

[Download Puzzle Solutions Here](#)

Carnival Word Find



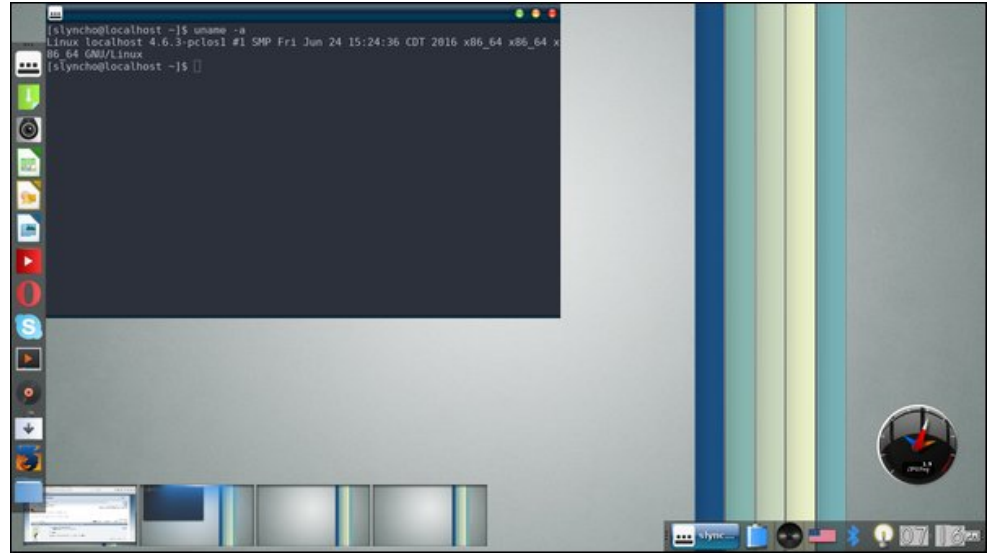
1. ice with flavored syrup on it
2. batter, fried and covered with powdered sugar
3. a person wearing a funny outfit and doing crazy things
4. the place to get your snacks
5. a great snack when salted and buttered
6. fluffy spun sugar
7. the _____ are the only time it's ok to hit another vehicle
8. an animal show
9. fun when it stops at the top
10. funny reflections
11. frozen dairy product
12. how does he keep all those objects up in the air?
13. the fast, scary ride!
14. candy-covered fruit
15. circular ride on a horse or lion or zebra!

[Download Puzzle Solutions Here](#)

More Screenshot Showcase



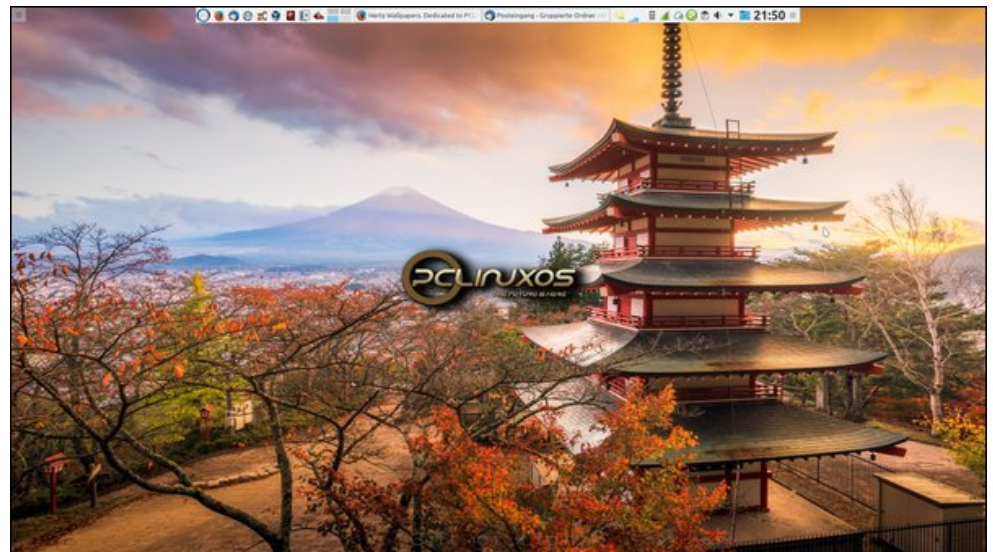
Posted by Meemaw, July 11, 2016, running Xfce.



Posted by daniel76, July 18, 2016, running e20.



Posted by HERTZ, July 23, 2016, running LXDE.



Posted by tbschommer, July 10, 2016, running KDE.