MUUGLines

The Manitoba UNIX User Group Newsletter

Volume 37 No. 1, September 2024

Editor: Trevor E Cordes

Next Meeting: September 3rd, 2024 7:30pm

In Person and Online

Mini-Presentation: Fly light!

Kevin McGregor will provide a gentle introduction to the lighttpd web server.

“lighttpd (pronounced */lighty/*) is a secure, fast, compliant, and very flexible web server that has been optimized for high-performance environments. lighttpd uses memory and CPU efficiently and has lower resource use than other popular web servers. Its advanced feature-set (FastCGI, CGI, Auth, Output-Compression, URL-Rewriting and much more) make lighttpd the perfect web server for all systems, small and large.”

Desktop App Mini-Presentation

For our mini-presentation of a “desktop app”, Brad Vokey will do a quick demonstration of his favorite GUI text editor: Sublime Text - “Text Editing Done Right”. Sublime Text (sublimetext.com) is a fast, mature, light weight, beautiful, customizable, expandable, cross platform, “shareware” text editor. Brad has previously written about Sublime Text in the MUUG 2021-06 newsletter and he will now give us a quick intro into some of it's features.

If you show up in person you will be treated to more beverage choices than we’ve offered in over a decade: coffee, tea, and pop, as well as cookies. And parking is free, copious, safe, and just a handful of feet from the door.

To attend via internet, check and refresh the following link after 7:00pm. There is no need to create an account in BBB, nor login. Just enter any name as your screen name and hit **join**.

`https://muug.ca/meet`

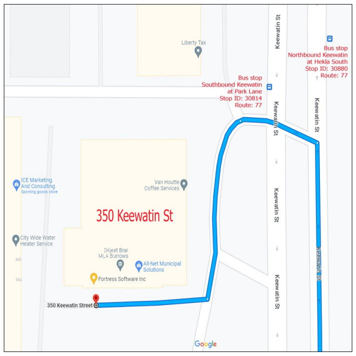
The latest meeting details are always at: `https://muug.ca/meetings/`

Where to Find the Meeting

We are continuing to meet on the first Tuesday of every month.

Fortress Software Inc.
350 Keewatin St – Unit #2

Doors will open at 7:00pm. Meeting starts at 7:30pm.



If driving, enter the lot using the most north-eastern entrance (far right in the top picture) and drive around to the south west corner of the building (see route in map detail). You can use any of the free, ample, and safe parking spots that say “reserved” or “MUUG” in front of units #1 through #4 before entering into unit #2. Look for the sign over the door!

Bus stops #30814 and #30880 (route 77) are only 150 meters away. The last bus leaves for Polo Park at 10:15 pm and for Garden City at 10:31 pm. Logan Ave. bus routes #19, #26, and #27 are a 600 meter (8 minute) walk to the south.

Door Prizes

There are a couple of newer batches of e-books you may want to peruse as a possible door prize choice this month: both with a veteran Linux administrator focus.

*NIX-themed stickers continue to be offered as a couple of prize draws, as well as *Communications of the ACM* magazines graciously donated by Dan Keizer.

And, as usual, one lucky winner can select an e-book graciously donated by Michael W. Lucas of Tilted Windmill Press (see advertisment on last page).

Everyone is encouraged to enter the door prize draw: both in-person and virtual attendees. You don’t have to be a member to enter, but members have exclusive access to the first draw. Members also get the privilege of not having to be present (physically or virtually) for the draw in order to claim their prize.

New this season: physical prizes can only be selected by physically present attendees. If you attend online you’ll have to select from one of the e-books if you win a door prize.

MUUG Summer BBQ a Huge Success

Attendees at our BBQ In The Assiniboine Park were treated to a great time, and with a record 21-person attendance!

The slight rain departed just in time, the early date kept the wasps away, and the mosquitoes were kind enough to buzz off until dusk. Perfection!

Thank you to all of the members who brought food to share, and to every attendee. With such success MUUG is sure to stage another one in 2025; so if you missed it this time, join us then!

25 Year Old Firefox Bug

Firefox 128, released in July, fixes a bug that was first reported 25 years ago! Bugzilla bug #33654 “*TEXTAREA incorrectly applying ROWS= and COLS= (horizontal / vertical scrollbar extra space, with overlay scrollbars disabled)*”.

It would seem kludgy code to account for the optional appearance of scrollbars in a textarea input field was causing the actual column and row counts to be off by one, or two (or eighteen!). Font appearance was also going wonky.

Who says open source projects aren’t responsive to user needs! Oh… wait…

Before it was closed the bug received 27 duplicates, 103 votes and 91 CCs (interested parties).

Better late than never?

The whole sordid affair is chronicled here:

https://bugzilla.mozilla.org/show_bug.cgi?id=33654

Can’t Shrink a Root FS Remotely! ?

Many filesystem types can be shrunk. This has been possible for quite a long time now. Unmount the filesystem and run the appropriate shrink command on it. But no filesystem this author is aware of allows online shrinking; so that umount is a critical first step.

But what about a root filesystem? It can't be unmounted while the box is on and booted. So we all know the solution to that: get a live CD or sysresccd or Knoppix, and boot from that, and then shrink the now-offline/unmounted filesystem. Easy peasy.

But what about if the box is remote, and difficult or impossible to get to? What if it's hosted in some rack somewhere in a different country? What if your dedicated hosting provider provisioned your box incorrectly and it'll take them a week to reprovision it? Surely you are doomed in that case!

Not so fast… Andrew Wood seems to have found a way… in 2007!

The Cliff's Notes version is: you make a "fake" temporary root fs on a tmpfs mount, then copy key/critical directories from root and usr and var into the tmpfs. Then you "pivot the root" with pivot_root to the tmpfs root. Use fuser to id and kill any ps still using the original root. Then you can unmount it! With the system still running like nothing is happening! <Minds Boggling> Do your resize, pivot back, and discard the tmpfs.

Voila! You just did the impossible. Almost as good as this author's patented in-place remote upgrade of a Linux system from 32-bit to 64-bit <wink>. Yes, with Linux, virtually anything is possible!

Original version:

https://www.ivarch.com/blogs/oss/2007/01/resize-a-live-root-fs-a-howto.shtml

Updated version for modern systems (use this):

https://unix.stackexchange.com/questions/226872/how-to-shrink-root-filesystem-without-booting-a-livecd/227318#227318

Linus Pooh-poohs ZFS

> *"Don't use ZFS. It's that simple. It was always more of a buzzword than anything else, I feel, and the licensing issues just make it a non-starter for me."*

-- Linus Torvalds

To be fair, his audience for the comment was distro and kernel developers and maintainers. But still, to some people, *them's fightin' words.*

Though ZFS was made open source in 2003, it is released under the CDDL 1.0 license, not the GPL 2.0 that Linux uses. Since CDDL is not "fully compatible" with GPL, it can't be included with the kernel. Thus all of the extra hoops people who want to use ZFS on Linux have to jump through.

Linus expounds further:

> *"Other people think it can be ok to merge ZFS code into the kernel and that the module interface makes it ok, and that's their decision. But considering Oracle's litigious nature, and the questions over licensing, there's no way I can feel safe in ever doing so."*

But there is an out to end the pain and anguish of ZFS-lovers everywhere (emphasis mine):

> *"And honestly, there is no way I can merge any of the ZFS efforts* ***until I get an official letter from Oracle that is signed by their main legal counsel*** *or preferably by Larry Ellison himself that says that yes, it's ok to do so and treat the end result as GPL'd."*

Some distros have decided to take the risk anyhow. Ubuntu has allowed root ZFS partitions since release 19.10. Maybe unlimited pockets to fight potential legal battles had something to do with that?

> *"The benchmarks I've seen do not make ZFS look all that great. And as far as I can tell, it has no real maintenance behind it either any more, so from a long-term stability standpoint, why would you ever want to use it in the first place?"*

So Linus is definitely not a ZFS fan!

N.B. The muug.ca server has used ZFS successfully for many years to hold the massive distro (etc.) mirror data.

https://itsfoss.com/linus-torvalds-zfs/

Why Can’t My Obscure Window Manager Be Ported to Wayland?

This author is a huge Sawfish Xorg Window Manager (WM) fan. It is, in some ways, the closest match to late SunOS / early Solaris and AIX CDE WMs that this author first fell in love with. Even better, it is fully customizable and programmable in Lisp! Easy to use, yet powerful, config GUIs provide almost all the tweaks you would normally need.

But all the buzz (for years now) is around Wayland. The Lennart Generation has decided Xorg is bad, because reasons. Great, so give me Wayland with my favorite WM and I’ll be on my way.

Not so fast! Wayland doesn’t have *WMs*. The concept doesn’t exist there. What Wayland has is *compositors*. A compositor is like an international space station, while a WM is like a teenager’s hobby rocket.

RobertZenz provided the best answer yet to this issue in a github comment in 2023, and it’s so good, it’s worth quoting nearly in full (emphasis mine):

> *You see, every Wayland advocate will happily tell you that “Wayland is just a protocol”, and that's true. There's a “reference” compositor which is called Weston, which is bare-bones, and everything else is left to everyone else. To “support Wayland” one must implement a complete Wayland compositor, that includes implementing and handling of rendering, hardware acceleration, input, clipboard, window management, screenshots, capturing of the desktop for streaming and so on. In short,* ***every Wayland compositor must basically implement all the functionality which is currently provided by the X.org server****. In my opinion, the big Wayland motto is “It's somebody elses problem now”, because that is exactly what it is, Wayland is just a protocol, everyone must implement everything. If there is no Wayland protocol for something, then the compositors are left to their own devices to implement that functionality, which means that all the Wayland clients (GUI frameworks) must implement that compositor specific function, and so on. Additionally, GNOME slightly disagrees on a few key functionalities (window decorations), which makes it a little bit worse for everyone.*
>
> *So the questions “when will Wayland support Sawfish?” or “when will Sawfish support Wayland?” are incorrect, the correct question is “when will someone write a complete Wayland compositor (including rendering, hardware acceleration, input, clipboard, window management, screenshots, capturing of the desktop for streaming and so on) which will behave like Sawfish?”...I guess we all know the answer to that one.*

And the above words apply equally as well to any Xorg WM project that does not have the resources of GNOME and KDE to actually create a full compositor.

Basically, if Xorg goes away, everyone must use one of the (very few) existing Wayland compositors. And the chances of one of those generally dumbed-down interfaces supporting a quirky feature you find critical is slim to none... And slim just kernel panicked.

But all hope is not lost. The general consensus on the sawfish mailing list is that distros will support Xorg for many years to come. Crazy you say? Why you ask? Because BSD can never do Wayland. Yes, BSD may be the saviour of Obscure WM Lovers everywhere. Wayland relies on way too many *kits and Linuxisms to ever be accepted by the BSD purists.

As long as an app wants to run on BSD too, they will compile in Xorg support. And as long as apps have Xorg support, why wouldn’t distros do likewise?

If/When distros finally start shedding Xorg, there will inevitably be *rebel distros* just like there was when systemd took over. Devuan (and others) to this day provide a Linux that is systemd-free. Something similar will pop up to continue making Xorg and sawfish a viable option.

`https://github.com/SawfishWM/sawfish/issues/53`

A big thanks to Les.net for providing MUUG with free hosting and all that bandwidth! Les.net (1996) Inc. is a local provider of VoIP, Internet and Data Centre services. Contact **sales@les.net** by email, or +1 (204) 944-0009 by phone.

MUUG would like to thank Michael W. Lucas for donating one of his ebooks every month as a door prize. You can view and purchase his tech books here:

https://www.tiltedwindmillpress.com/product-category/tech/

Help us promote this month's meeting, by putting this poster up on your workplace bulletin board or other suitable public message board:

https://muug.ca/meetings/MUUGmeeting.pdf