# MUUGLines

The Manitoba UNIX User Group Newsletter

Volume 39 No. 1, September 2026 Editor: Trevor E Cordes

## Next Meeting: September $1^{st}$, 2026 7:30pm

## In Person and Online

## Main Presentation – A Mystery?

As of press time, we still don't know if the presentation will be by Kevin McGregor, presenting a Daemon-Dash style on a topic TBD, or Wyatt or Brad, doing a /bin Diving presentation.

Check the `muug.ca` website for updates, or just show up at the meeting and be surprised!

## Where to Find the Meeting

**Winnipeg Senior Citizens Radio Club**
**598 St. Mary's Rd. Winnipeg, MB.**

MUUG has moved to a new location for the 2025-2026 season. Winnipeg Senior Citizens Radio Club (WSC) has graciously offered to host MUUG


meetings going forward. Several current and past MUUG members are also members of WSC. MUUG President Wyatt Zacharias will be our WSC liaison and duty officer.

WSC is located at 598 St. Mary's Road, $2^{nd}$ Floor, a block South from the St. Mary's / St. Anne's fork, on the West side beside the Miller's Meats. It's in the old fire hall, so you really can't miss it! Look for the red doors.

The meeting space is on the **$2^{nd}$ floor** and there is no elevator, so attendees will have to be able to climb 1 flight of beautiful character style stairs. Apologies to anyone requiring accessibility. It is also possible to attend via online videoconferencing: `https://muug.ca/meet`

Meetings are on the first Tuesday of every month except July & August. You can attend in person or online via our MUUG videoconferencing system.


Beverages and cookies or treats will be provided.

Doors will open by 7:15pm. Meeting starts at 7:30pm, and usually adjourns between 9:00pm and 10:00pm.

To attend online, check and refresh the following link after 7:15pm. 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/`

## CVEs Gone Wild

Maybe you've noticed that your boxes are telling you there are updates, often critical and reboot-required ones, almost every day lately? Things like chromium are having 10-100 CVEs nearly every day. And the venerable kernel is seeing a lot of activity too, which frankly is the worst because that often warrants a reboot.

What's going on?

The main driver of this insanity is AI vulnerability scanners. People and companies are using AI to pore over open source code bases looking for bugs that follow known paradigms. It can do it way faster than a person can, and no programmer actually likes looking through 50k of lines making sure every malloc is working properly.

As for the Linux kernel, they have changed the way they issue CVEs, basically taking over their own CVE numbering. They also changed what they will issue a CVE for, and when, resulting in many more than we are used to from the project.

Should you worry? Well, yes, you should always be scanning your distro's CVE mailing list and/or applying every update daily. The tricky bit is knowing when to reboot, if you're the type who has 1000 things open, or is running production servers that need high uptime.

Linus has said he expects the level of insanity to eventually die back down to normal levels. We can all cross our fingers.

## Stop That OOM!

Ever had a program start eating up all of your RAM? The system bogs down as swap thrashes. If you're very lucky the oom-killer (oom==Out Of Memory) daemon will kill it off. If you're unlucky, which is often, it won't, or it'll pick some Xorg or Wayland process to kill. And that usually freezes your graphical session and likely the video driver, putting you in a bit of a bind. Firefox especially seems prone to this.

Did you know there's a way to run a program that will limit its physical RAM usage, thus putting a stop to the aforementioned symptoms. Before you say it, nope, it's not ulimit! ulimit only operates on *virtual memory*, not on *physical memory*. This means it won't work with browsers, which often pre-reserve many times more virtual memory than you have physical. And most goes unused and never allocated out of real RAM.

Ah, how about firejail you say – as you were there for the wonderful firejail MUUG presentation the other year! Nope, alas firejail cannot do it either.

It turns out the key is (groan) systemd-run. It uses the newfangled kernel slices and cgroups and other opaque voodoo kernel wizardry to effect the limits you want. And you can combine it with the firejail setup you're already using (you are using it, aren't you?) to basically cover all of your bases.

Make a little script called firefox in your ~/bin or usr/local/bin, or some other place that is earlier in your path than /usr/bin. In it put:

```
#!/bin/sh
exec systemd-run -t –user \
--same-dir --collect --wait \
-p MemoryMax=25G \
-p MemorySwapMax=2G \
-p MemoryHigh=24.5G \
firejail /usr/bin/firefox "$@"
```

(If you're not using firejail, just omit the word *firejail*. And don't forget chmod 755.)

This works great on this author's 32GB RAM system. However, you do have to make sure you don't actually open so many windows or tabs open that you start hitting this limit. If the limit is reached, the kernel will just kill the process, and probably not nicely! While that is desirable when a program is "running away", it's no fun when everything is working fine and all of your browser windows suddenly close.

If you want to see that it's actually doing all of this fancy stuff while it's running, you can use systemd commands to query it. First, find its Id:

```
systemctl --user show "run*" \
--property=Id,MemoryMax,MainPID
```

Find the one that has MemoryMax that looks like what you set it to previously. Remember that Id, which should look something like *run-p41252-i382192.service*.

Then see more details than you'd ever want to know about:

```
systemctl --user show \
ID | grep -i mem
```

Replace ID with your Id from the previous step. And omit the grep pipe if you want to see *everything*.

```
MemoryAvailable=23102337024
MemoryCurrent=3204337664
MemoryHigh=26306674688
MemoryMax=26843545600
MemoryPeak=4923523072
MemorySwapCurrent=364236800
MemorySwapMax=2147483648
MemorySwapPeak=847396864
```

As you can see, in my case, firefox is currently using only 3.2GB actual RAM, and 364MB swap, and the desired maxes are in place. Cool!

## Dnf Leaves

No, don't worry: dnf (the Red Hat-based distro package manager) is not going anywhere. *Leaves* means end-of-tree nodes – or packages that don't have anything depending on them. So firefox is a leaf, as would be most games. On my box there are 983 leaves!

Just type: dnf leaves

This is great for quickly seeing packages that are ripe for removal, to clear up space or make distro upgrades faster. You can remove any of them without worrying about dependency hell trying to delete 214 other packages. And dnf is smart enough to auto-(suggest-)remove any packages further up the tree that were installed just for that leaf.

## Spot The Hard Drive Deals

Why oh why did someone not tell me about this site sooner? diskprices.com is a simple web page that shows a table of currently available hard drives sorted by lowest price per GB. It allows you to specify some parameters, like rust or SSD, internal or external.

If you're building a big RAID array, this can help you plan your purchase to get more space for your buck. As of printing time the best price for internal rust is $0.026/GB for an 18TB WD "video" drive. The best price for a "normal" drive is $0.027 for a 6TB HGST. A Seagate Exos 24TB drive isn't far behind.

These drive sizes are getting a bit nutso – and that's a good thing!

`https://diskprices.com`

## Crashing Systemd

Whatever you do, no matter how tempted you are, or how good a reason you think you have, do **not**, while you are logged in graphically, run either of:

```
systemctl restart systemd-logind
systemctl restart dbus
```

While both sound cool, because maybe you just changed a setting in their conf files or upgraded something, do not do it! Both will instantly lock your keyboard and mouse, whilst leaving all of your apps visibly running (i.e. not frozen!) on the screen.

Switching to another virtual terminal will **not** work either. But `magic sysrq` will! And you can use `magic sysrq keys` to get the box to a state where you can type commands and restart your X without a reboot.

## The Incredible Shrinking /boot

Do you have a separate /boot partition, as many distros create by default? Are you running a box that has been upgraded 10-43 times over decades? Has your update/upgrade system been telling you that it can't do its job because your /boot is too small?

Take Fedora for instance. Originally FC1 made a 100MB boot partition. Fedora 13 500MB, 24 1GB, 43 2GB.

Why? Because the kernel initramfs boot image that contains all of the drivers and critical early-boot stuff keeps getting bigger and bigger! And most package managers keep 3-4 versions of the kernel by default!

So what do you do when this occurs? (And it will occur!) There are a few options, from easiest to hardest.

1. If you have a dracut-based system (like RH-based distros, and many others), you can tell dracut to omit unused junk from the initramfs.

```
nano /etc/dracut.conf.d/disable-
nouveau-to-save-initramfs-
filesize-space.conf

omit_drivers+=" nouveau "
```

The above eliminates nouveau nvidia video card drivers. If you do not have an nvidia card, or are using the nvidia binary drivers, then you do not need these huge drivers.

Run dracut -f to regenerate the initramfs file. On this author's system it shrunk the files from 134MB to 28MB! Who says you need a bigger /boot? Mine is still only 400MB from FC3-ish era. Just don't screw up which drivers you think you need! Or your system will be unusable when you boot into that kernel.

2. Tell your package manager you no longer want to have a *rescue* kernel image. No one really uses it anyhow. Then delete the matching *rescue* files from /boot.

3. Tell your package manager (like dnf) you only want to keep 2 kernels around instead of 3. This is a bit riskier, as you lose the ability to go back very far in time should the kernel you were just running and the one just installed both be wonky.

4. Resize your /boot partition. This is somewhat easy if you are sitting there with the box: boot into a gparted boot CD/USB and use its graphical tool to juggle partitions around and make /boot bigger. Modest risk, so make a backup first!

If the box is off-site, then resizing becomes trickier, since you usually won't have free disk space right after the /boot partition, which is usually first. However, anything is possible with Linux, and you can do all sorts of wacky and insane partition meddling even on running systems. Caveat emptor!

**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

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.

Thanks to Linux Mag for providing MUUG a half price magazine subscription for our door prizes.

Linux Magazine is your guide to the world of Linux and open source. Each monthly issue includes advanced technical information you won't find anywhere else including tutorials, in-depth articles on trending topics, troubleshooting and optimization tips, and more! Subscribe to their free newsletters and get news and articles in your inbox.

https://www.linux-magazine.com/

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/