



Linux admin tips

This is aimed at people who administer Linux systems, but whom are not (yet) Linux geeks and have some Linux based/related systems, i.e., usually coming from another platform such as Windows or NetWare. Certainly getting some formal Linux training will help, but have found that much of that training very quickly assumes that all computer techs are programmers which is not the case.

This document is from the SUSE distro perspective, usually with Micro Focus's Open Enterprise Server (OES) and/or GroupWise running on it but certainly can help for many other applications. Assumed is that you are in an expandable terminal window that you can expand past 24 lines of 80 characters, such as when using a tool such as PuTTY. Commands will be in *italics* and should be typed in the same case as given as the examples as Linux is case sensitive (lower case vs UPPER CASE), and sample output will be in a `fixed font`.

The first three commands are the ones I do most times to check on a server, basically asking these questions:

- How long has the server been up?
- Is any process using way more CPU or memory than normal?
- Are we getting tight on memory?
- Are we getting tight on drive space?
- Certainly there are system management systems that are available that watch for these, but they aren't always an option due to their cost and available budget for them, nor are they immune from failure. Checking these periodically and knowing what is normal for your systems is a very good habit.

top All basic Linux training should have shown you the Top tool that lists the processes running, though I have found that they tend to either gloss over it, or hit you with ALL the bells and whistles. The basics to look for starting from the top of top are:

- 1st line: (also available with *uptime*). How long has the server been up. Are any of the load average numbers getting close to the number of CPU cores?
- 2nd line: Making sure there are no zombie or stopped processes.
- 3rd line: How much of the CPU is being used right now.
- 4th and 5th lines are similar to the free command explained below.
- The remainder shows the top processes, with the default sorting by CPU usage with highest to the top(as in show the top busy processes). You can change the sort order:
 - **M** (Shift-m) to sort by memory use,
 - **T** (Shift-t) to sort by how long the process has been running,
 - **P** (Shift-p) to get back to the default sort by amount of processor used.
 - **c** to toggle, seeing the full command for a given process.

The top line shows you how long this server has been up. Occasionally a process can gradually take more and more memory or CPU cycles, aka memory leaks or stuck threads, and top is where you would likely see it first if you are checking regularly.

free -m What memory the server sees and how much is actually used. Confirms that there really is the amount you expect running which catches memory failures and theft. Shows how much of your swap space is used. If you are running out of swap space, or it is really heavily used, increasing assigned/installed RAM may be in order. You don't want to endlessly grow your swap space unless poor performance and wasted electricity is acceptable (spinning hard drives takes more energy, and a longer time to do anything than RAM).

df -h What drives your system has and the percentage used. Check regularly to make sure you are not running tight anywhere as it can be bad to run out of space in root (/) just like it is bad on C: on a Windows box

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/sda2	18G	8.2G	8.7G	49%	/
devtmpfs	2.9G	120K	2.9G	1%	/dev
tmpfs	2.9G	144K	2.9G	1%	/dev/shm
/dev/pool/DATA	900G	710G	191G	79%	/opt/novell/nss/mnt/.pools/DATA

Root (/) looks good, but we will have to periodically watch the DATA volume.

To figure out what is using space on a drive, use **du -hx --max-depth=1** This will give you a summary of space used in each directory/folder from the level you are running it at, but only for the mount point you are on. Then for directories that look too big, you can **cd** into them and repeat until you have found what is taking up the room. Note that this command can take a bit of time to run, so be patient with it. It is a good exercise to do, to know your system.

One problem is where some log files grow way too fast for some reason and the above process with 'du' will find that fast enough. If the logs in question are already being bziped, then either remove them or expand space to allow for a year's worth. Otherwise when you have a log file that is huge (greater than 20MB and growing) investigate logrotate. While many logs such as apache's are set with logrotate as a default, there are some that are perceived should never grow much so haven't had a logrotate process setup for them. Usually just look at the existing rotation files in /etc/logrotate.d, make a copy of the one that is closest to my needs, and edit it accordingly.

Operating Systems, do not as a rule, automatically cleanup old files in their TEMP folders so you can get quite a lot of files building up on some systems. To clear /tmp and any others you need old files cleared, edit **/etc/sysconfig/cron** on SLES/SLED.

More details are available at the link listed below.