

Practical Hands-on guide to Linux

Shantanu Desai, IIT Hyderabad

What this talk will cover (or not)

- This is not an exhaustive introduction to Linux
- This talk will only cover the most important skillset one needs to know to deal with Linux systems and do analysis
- Expect most of you to already know most of this stuff
- Assumes all of you have logged in to a linux system at least once.
- The command will also work on non-Linux flavors of Unix.

This talk does not cover

- Linux system administration
- Software installation
- Difference between various linux shells : bash, tcsh
- Shell Programming (not recommended) - use Python or perl instead

How to find more information about any Linux command

`man <command>` or `<command> -h/--help`

or `google`

Environmental variables

- Default Environment variables are set upon logging and in your .bashrc file (or sometimes .login, .bash_profile etc)
- One can also write all these in a file and on command line use `source file`
- How to check all environmental variables : `env` (or `printenv`)
- How to print/display a particular environmental variable : `echo $VAR`

PATH environment variable

- Most important environment variable is PATH. It controls where system searches for executables. If a code is in a directory and that directory is not in contained your PATH variable you will get an error that command not found
- You can append new directory locations to your existing PATH by separating using colons `export PATH=${PATH}:<newdirectory>`
- Note that the order of directoros in the PATH env. variable is important. It looks for executables from the leftmost directory in the PATH variable

Related env. variable `LD_LIBRARY_PATH` , `PYTHONPATH`

How to find location of executable on your system

Eg. You want to know which version of `dspsr` `pinta` is using (or location of `python` executable used in your system)

`which dspsr` (or `whereis dspsr`)

In case you get a message that a particular command (or executable) is not found, it implies one of two things:

- That executable does not exist on the system.
- That executable exists, but the directory containing the executable is not added to your `PATH` environment variable.

Command line shortcuts

Ctrl e - go to end of line

Ctrl a go to beginning of line

Escape f - go to next word (after space)

Escape b go to beginning of current word

Up arrow (Trace command history) (

Ctrl -K delete the full line (ctrl Y does the opposite)

Ctrl w (delete previous word)

<Linux command> !\$ (prints last word of the previous command). Very useful

How to find files on your system

Two broad methods

1. `locate`
2. `find` `find` can also be combined with other Linux commands to carry out multiple operations : (very versatile)

Exercise: find all files owned by you

File permissions, groups etc

For full detailed listing about a file do `ls -l <filename>`

```
-rw-r--r-- 1 shantanu tvisitor 110299 Jan 13 22:45 J2124-3358_59196.452419_1460.gptool.summary.pdf
```

File permissions can be changed using `chmod`

For example to give group readable access `chmod g+r <filename>`

To find all groups you belong to `groups <username>`

To change the group of a file use `chgrp <newgroup> <filename>`

Moving, copying of files and directories

- `mv <oldfile> <newfile>` Used to move or rename files (or directories)
- `cp <oldfile> <newfile>` Creates a copy of old file
- `ln -s <oldfile> <newfile>` Creates a symbolic link to old file without copying the file

How to change directories

- `cd <dirname>`
- `cd ~user` goes to home directory of user. (`cd ~` or `cd` goes to your own home directory)
- `cd ..` goes one level up directory tree
- `cd -` goes to previous directory where you were. Very useful

Copying files from one machine to another

`scp <filename> <ipaddress>:<path on remotesystem>`

(to copy a file from current machine to remote system)

and

`scp <ipaddress>:<filename on remotesystem> .`

(to copy a file from remote system to current machine in current director)

(in InPTA we use `scp -pr` to copy the raw files to preserve the datestamp and recursively copy the directories)

How to use wildcards while transferring files from remote machine

`scp remotemachine:<directory>:*fits .`

How to check if a machine is accessible

`ping <ipaddress>`

How to search for words in a file

grep example `grep <string> <filename>`

`man grep` gives many more options.

Exercise:

check the command to find the line number;

how to print both the file containing the match and the line containing the match

how to print all lines which don't contain a given match

How to combine multiple linux commands

Use “|” between two linux commands (usually)

Example :

`ls|grep pdf` (will look for all files containing pdf)

How to check and stop running jobs etc

How to check all user intensive jobs : top

How to check all jobs running by shantanu `ps -ef|grep shantanu`

How to kill running jobs

- Get process-id of the job using `ps -ef|grep <user>` or `ps -ef|grep <jobname>` (usually second column of the above output)
- `kill -9 <processid>`

How to kill ALL jobs by a user

`pkill -u shantanu`

(Use only in case of emergency as this will kill all jobs and also log you out)

How to ensure jobs run even after you are logged out

1. `nohup <job>` followed by `ctrl z - bg`

2. Alternately, one can use screen. I.e. open a screen session using `screen -S <screenname>` and run your job in that session followed by `ctrl a d`

Next time you login `screen -x <screenname>`

Screen won't work if the program uses an x-term and produces graphics on screen

Diskspace, memory, quota etc

How to check available free disk space

`df -h`

How to check disk space usage

`du -sh *`

How to check quota (only on tapti)

`quota -sv`

How to check available memory on your Linux system

`free -g`

How to create/uncompress a tar ball

Sometimes you download a tar file and want to uncompress it

`tar xfvz filename.tar.gz` (z option is need only if the file is a compressed file)

How to create a tarfile

`tar cfz filename.tar <listoffiles you want to compress>` (you can use wildcard)

Other very useful Linux commands/utilities

- **awk** Utility for doing arithmetic operations on files
- **sed (tr)** Utility for doing string operations on files (replace words, characters)
- **file** Check if a file is a binary or ascii file or symbolic link
- **wc** count no of words, lines etc (**wc -l** counts no of lines)
- **head** Prints file contents from the beginning
- **tail** Prints file contents from the end
- **sort** Sort files (both numerically or alphabetically)
- **cat** Concatenate files horizontally
- **paste** Concatenate files vertically
- **split** Split file horizontally
- **tac** Display a file from the end
- **uniq** report (or omit) duplications
- **join** Combine multiple files based on common fields
- **seq** Generate numbers
- **nl** Add line numbers to every file and print on screen
- **gzip** , **compress** compress a file
- **Gunzip** Uncompress a file (assuming file has been compressed with gzip)
- **wget** (or curl) download a file from a website (or ftp site)
- **tar** (create a tarball or untar a tarred file)

Some advanced utilities (which maybe useful)

How to parallelize linux commands :

- [Gnu parallel](#) (not installed by default on a linux system)
- [pexec](#) (never used it)
- [xargs](#)

How to see what executable is doing behind the scenes (to find bottlenecks in code)

[strace](#) [ltrace](#)

(However output of strace not trivial to interpret)

Conclusions and more information

Check out <https://sites.astro.caltech.edu/~srk/Unix/SRKUnix/UnixTools.html>

for more such utilities and important use-cases

One book to read more about Unix/Linux (although everything now available on google, stackexchange etc)

- Unix for Dummies by John Levine and Margaret Levine Young
- Chapter 1 of <https://prappleizer.github.io/textbook.pdf>
- https://en.wikipedia.org/wiki/List_of_Unix_commands

Some exercises to practice (used in InPTA)

- o Take a pipeline.in file from a different observation (but same set of pulsars & band) and modify it using the current date . use sed (or emacs) in one line
- o from pinta_summary output create a file which contains pulsar, MJD, gptool SNR, rfiClean SNR (use awk)
- o scp all files belonging to pulsars starting with J2 from gwbh machines after logged in to fs4
- o Find all the SNR.log files which do not have the expected 3 columns (use a combination of find and awk)
- o Calculate average, maximum and minimum rfiClean SNR of any pulsar observed in Cycle 41 (awk)
- o Find out which pulsars appear in SNR.log for a given MJD and band (use find and grep)
- o Remove non-ascii characters from a file.
- o Copy all DR files processed after a certain date in pinta v6.2 directory to a different directory. (find)