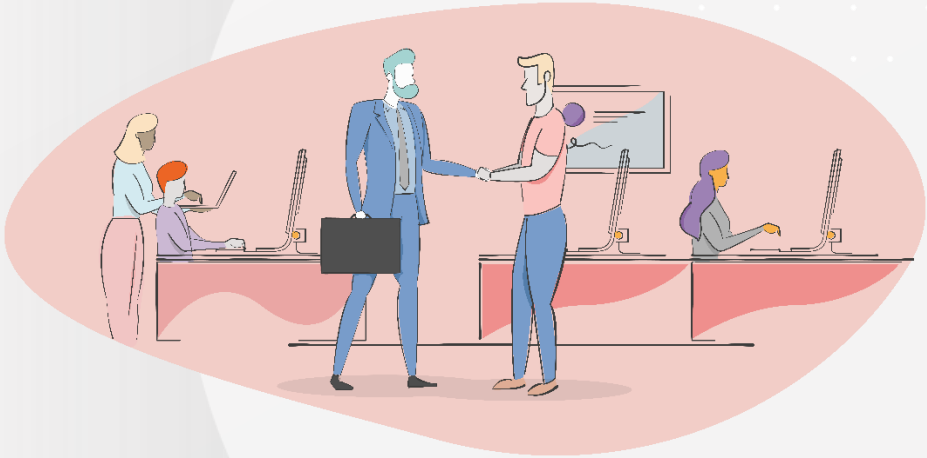
Linux for data engineers



L5 Data Engineer Higher Apprenticeship
Module 3 / 12 (“**Programming and Scripting Essentials**”)
Topic 1 / 8

rev. 1 (2024)

Ice breaker: Discussion

Building Careers Through Education

Everybody Matters. Stronger Together. Student, Learner and Client Centric. Trust and Respect. Embrace Change.

A bit of fun to start…

1. What's the most interesting thing you've read or watched this week?

2. If you could have dinner with any famous figure, who would it be and why?

3. What are you hoping to get out of this webinar?

Submit your responses to the chat or turn on your microphone

BPP

The real-world value of Linux

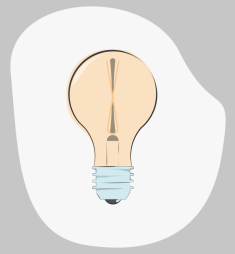
A real-world success story…

Walmart's data-driven approach to business operations and decision-making is largely supported by Linux-based systems supporting the back-end infrastructure of the business.

- The Linux file system underpins Walmart's datacentre storage
- Walmart Engineers use Linux CLI (command line interface) to automate tasks and extract insights from system logs

Walmart, a global retail giant, provides a compelling example of how Linux is used in data engineering*

**ASDA in the UK!*



Why do you think Linux is often chosen over Windows?

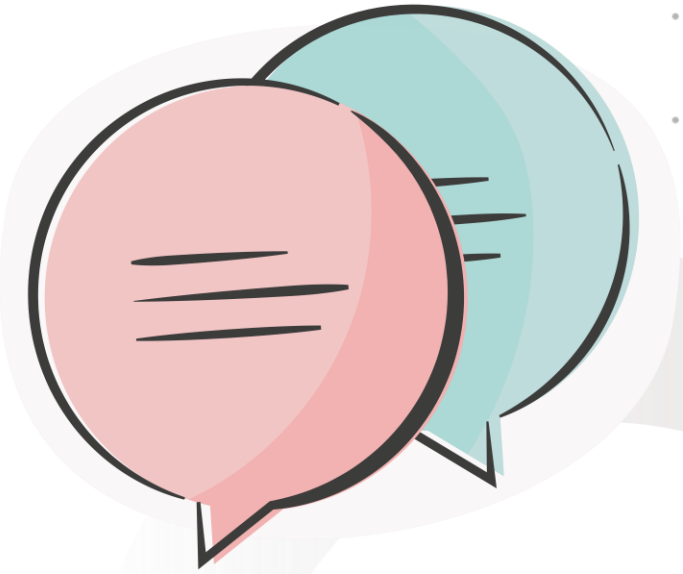
BPP

Spotlight on your experience

Discussion…

Consider the following questions and share your experiences:

- When have you used Linux to solve a problem in your role?
- Have you needed to use Linux-based tools for managing and analysing large datasets?
- Has anyone ever used scripting for automating tasks and log analysis for troubleshooting issues?



Submit your responses to the chat!

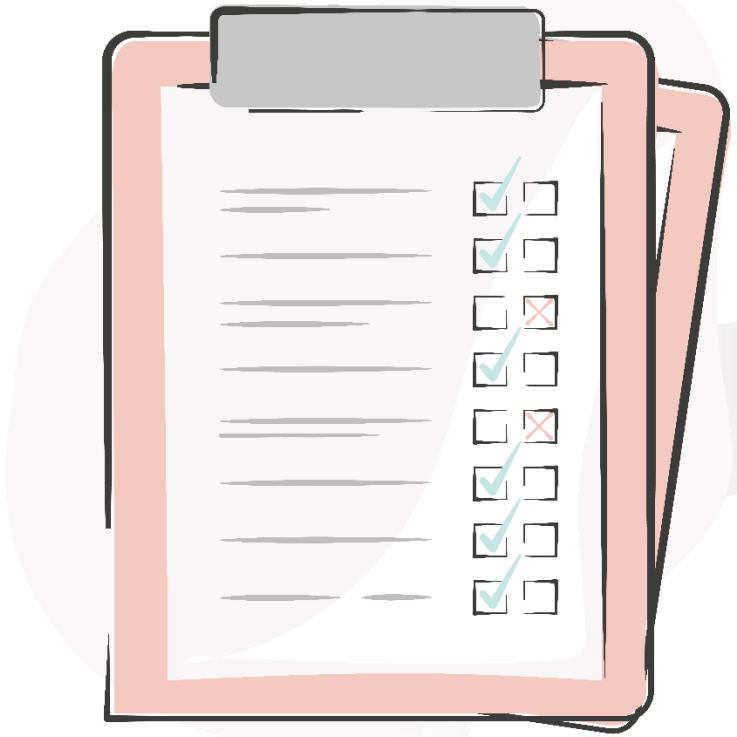
BPP

Webinar agenda

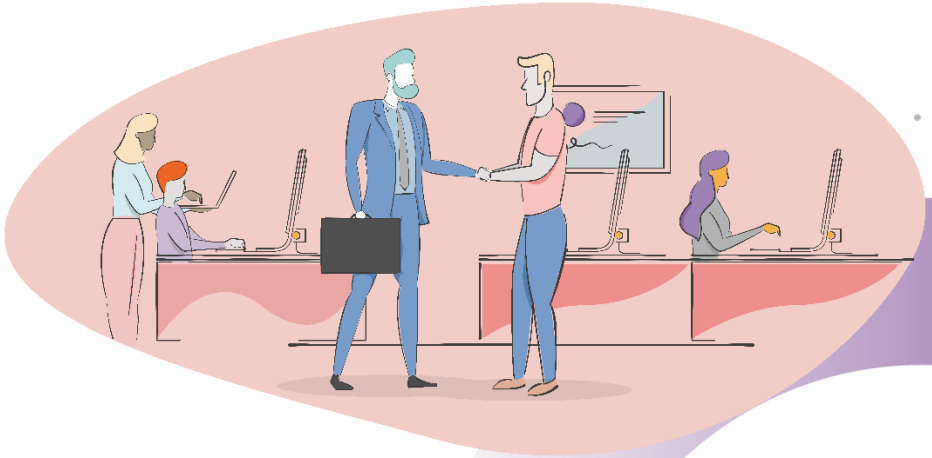
This webinar will cover the following:

- Where and why Linux is used
- How Linux works
- Practical Linux lab

Webinar length: 3 hours



Where and why Linux is used



Different OS philosophies

What is the difference between macOS, Windows, and Linux?

macOS

- Proprietary
- User-friendly
- Integration
- Security

Windows

- Proprietary
- Broad user base
- Historically reactive security

Linux

- Open source
- User freedom
- Customisation
- Community-driven

Introduction to Linux

What you need to know…

- Created by Linus Torvalds in 1991
- Unix based
- First version released in 1994
- FOSS (Free and Open Source Software)
- Can be modified as long as the source code is shared.
- Contains so called GNU Utilities, compatible with Unix (gnu.org)

Linus Torvalds

Linux has many ***distros***

Despite that "chaos", it is still seen as a single system.

- The distros share the same Linux kernel (core system)
    - Open-source code allows for easy bug review and fast issue resolution
- The distros share the same access model. Clear distinction between regular users and root enhances security.
    - *sudo* grants temporary privileges, reducing accidental changes
- The distros provide standard network protection tools, filesystem tools, scripting toole, etc.
- All Linux distros are less targeted by malware than Windows. **These makes them a great choice for server software.**

Testing different Linux distros

Which distro testing website is best for you? Try it out yourself!

Website	DistroSea	OnWorks	Instant Workstation	Shell Segfault
Internet access	No	**Yes**	No	**Yes**
Persistent storage	No	**Yes 10GB**	**Yes**	Yes 1.5 days
Pre-installed	No	**Yes**	**Yes**	**Yes**
GUI view	**Yes**	**Yes**	**Yes**	No
File upload/download	No	**Yes**	**Yes**	**Yes**
Mostly ad-free	**Yes**	No	**Yes**	**Yes**
Usually wait-free	**Yes**	**Yes**	No	**Yes (1 min)**

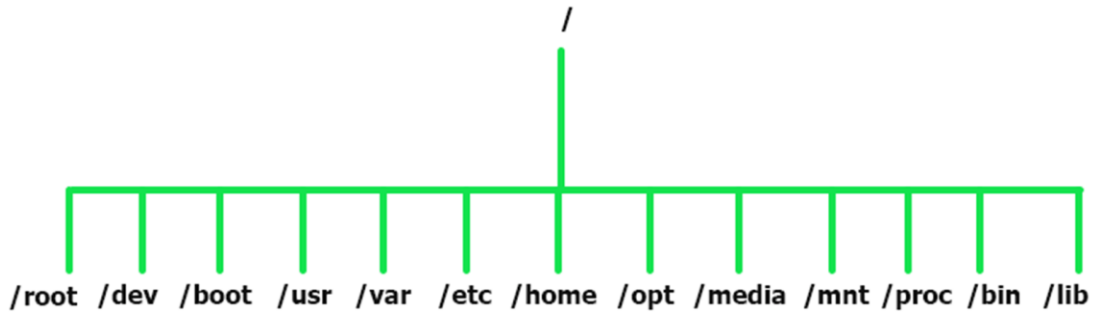
Free Websites that let you try out different Linux distros

Try out a new Linux distro and write about your experiences in your learning journal

Understanding the Linux filesystem

The directory structure

- The Directory Structure in Unix & Linux is a unified Directory Structure
- All the directories are unified under the “/” Root file system
- This structure is maintained irrespective of where the File System is physically mounted. **It is an abstraction.**
- All the directories are arranged hierarchically under the Root file system



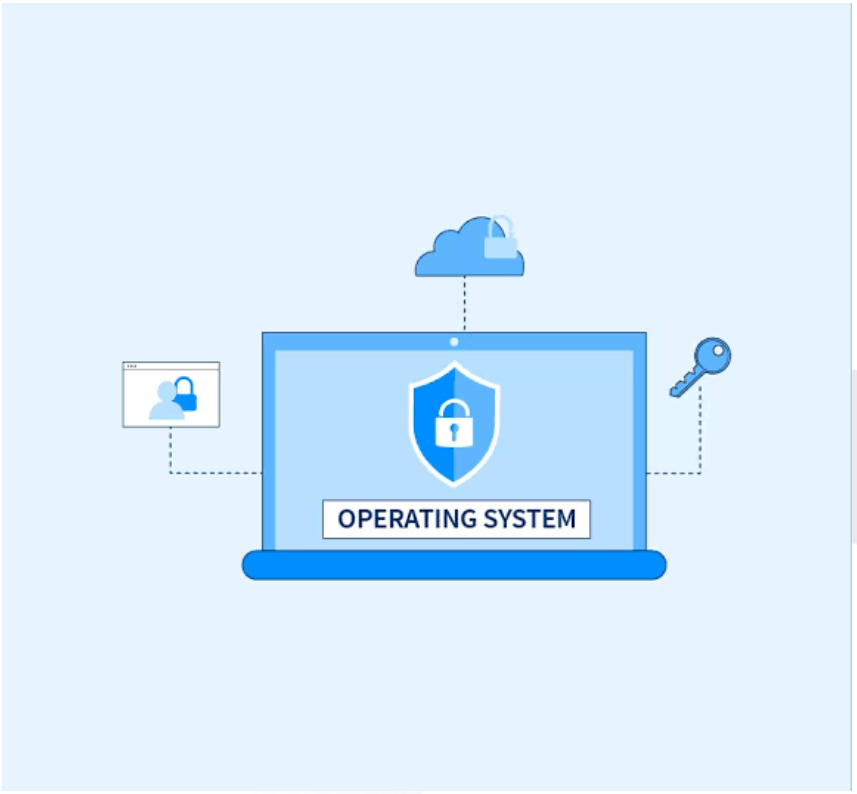
The Linux directory structure

Everything is a file

Building Careers Through Education
Everybody Matters.
Stronger Together.
Student, Learner and Client Centric.
Trust and Respect.
Embrace Change.

What you need to know…

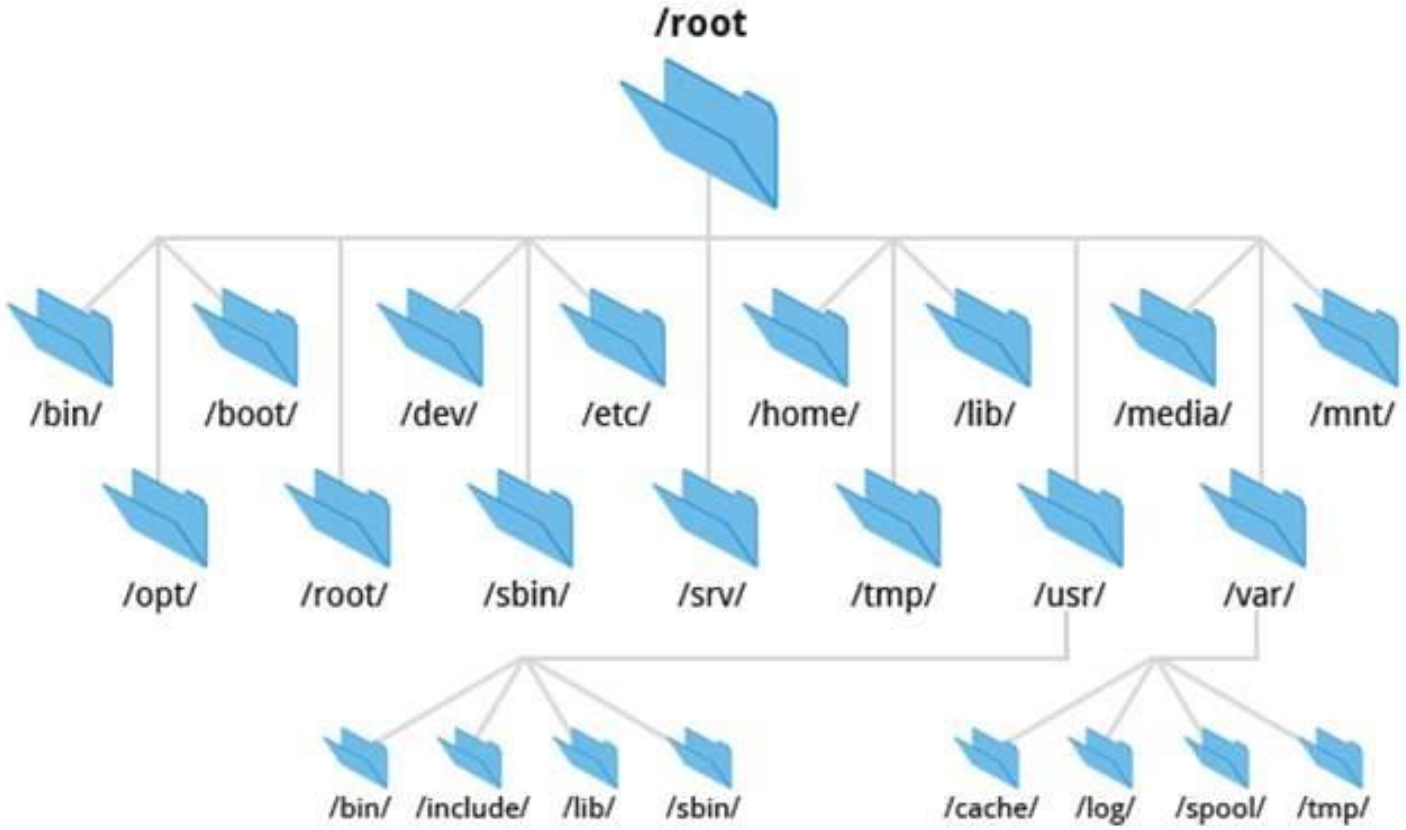
- in Linux everything as a file
- e.g. memory, device-drivers, named pipes, and other system resources
- hence why filesystem security is so important
- I/O to devices is via a “special” file
- e.g. /dev/cdrom
- have other special files like named pipes
- a conduit between processes / programs



BPP

Directories contain subdirectories

A close-up look at common directories and subdirectories



Common Linux directories

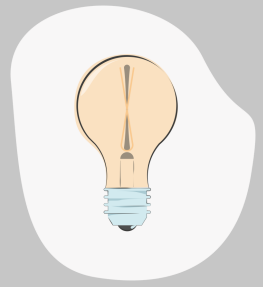
Directory is a special type of file

Every file in Linux can be one of the following four types:

- Ordinary File (contain data, info)
- Directories (hold files & other directories)
- Devices (for accessing the hardware)

- Links (Pointer to another file)

    - Hard Link
    - Soft Link



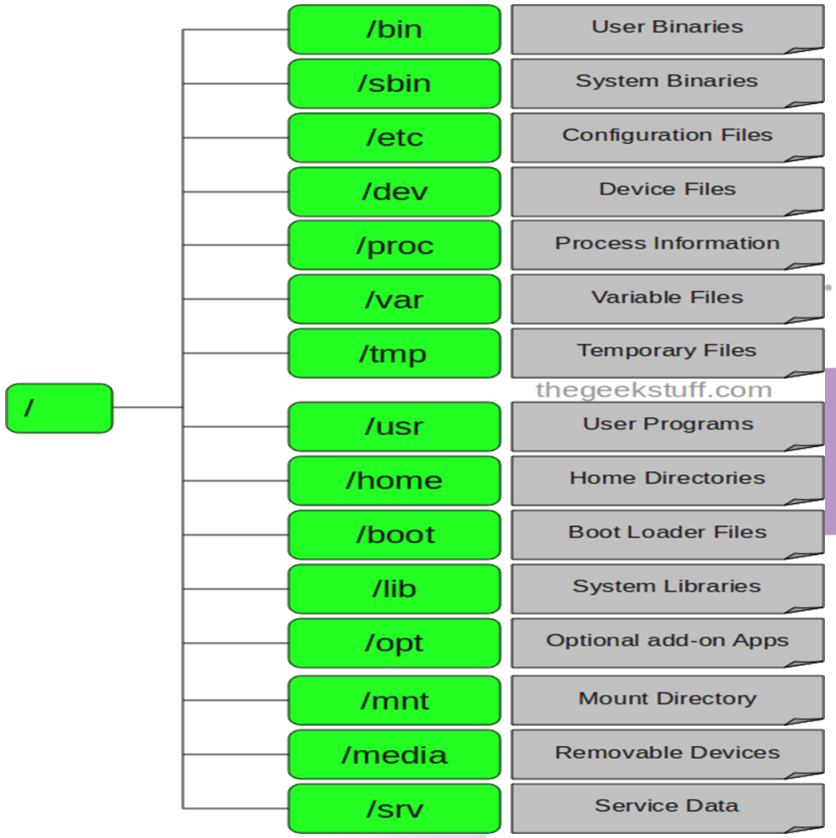
Every single file and directory starts from the / (root) directory.

Only root user has write privilege under this directory.

Special filesystem directories

A close-up look at important directories

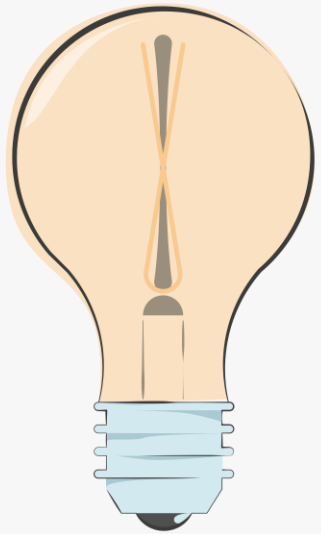
- /etc - Configuration Files
- /dev – Devices
- /tmp – Temporary Files
- /mnt – Mount Directory



Linux file system

Special filesystem directories

- /lib System libraries
- /lib64 System libraries (64bit)
- /lost+found Used by system to store recovered files after file systems check has been performed
- /proc Provides info about running processes



Helpful additional directories

/bin – User Binaries

- Contains binary executables.
- Common linux commands you need to use in single-user modes are located under this directory.
- For example: ps, ls, ping, grep, cp.

/sbin – System Binaries

- Just like /bin, /sbin also contains binary executables.
- But, the linux commands located under this directory are used typically by system administrator, for system maintenance purpose.
- For example: iptables, reboot, fdisk, ifconfig,

/var – Variable Files

- var stands for variable files.
- Content of the files that are expected to grow can be found under this directory.
- This includes — system log files (/var/log); packages and database files (/var/lib); emails (/var/mail); print queues (/var/spool); lock files (/var/lock); temp files needed across reboots (/var/tmp);

/usr *vs* /home

- In the original Unix implementations, /usr was where the home directories of the users were placed (that is to say, /usr/someone was then the directory now known as /home/someone).
- Currently, /usr is where user-land programs and data (as opposed to 'system land' programs and data) are. The name hasn't changed, but it's meaning has narrowed and lengthened from "everything user related" to "user system resources”
- /home is for home directories for all users to store personal files.
  - For example: /home/john

/boot – Boot Loader Files

- Contains boot loader related files.
- Grub files are located under /boot
- For example: initrd.img-2.6.32-24-generic, vmlinuz-2.6.32-24-generic

/opt – Optional add-on Applications

- opt stands for optional.
- Contains add-on applications from individual vendors.
- add-on applications should be installed under either /opt/ or /opt/ sub-directory.

Understanding Linux

A summary of the directory structure

File or directory	Details
Home directories	/root, /home, /username
User executables	/bin, /usr/bin, /usr/local/bin
System executables	/sbin, /usr/sbin, /usr/local/sbin
Other Mountpoints	/media, /mnt
Configuration	/etc
Temporary Files	/tmp
Kernels and Bootloader	/boot
Server Data	/var, /srv
System Information	/proc, /sys
Shared Libraries	/lib, /usr/lib, /usr/local/lib

BPP

Knowledge Check Poll

Which directory in Linux controls devices such as hard drives?

a) /root

b) /bin

c) /mnt

d) /dev

e) /media

Submit your responses to the chat!

Knowledge Check Poll

Which directory in Linux controls devices such as hard drives?

a) /root

b) /bin

c) /mnt

d) /dev

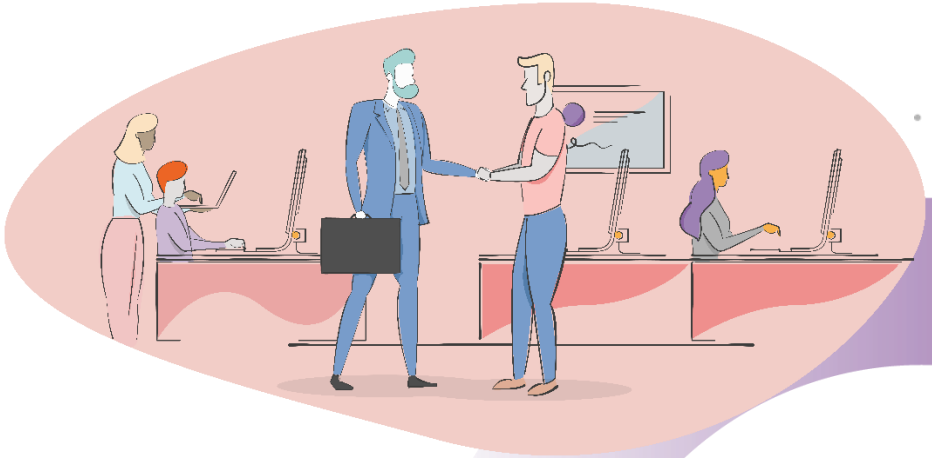
e) /media

Feedback: d – /dev, but /mnt and /media are also relevant – why is this?

Submit your responses to the chat!

BPP

How Linux works

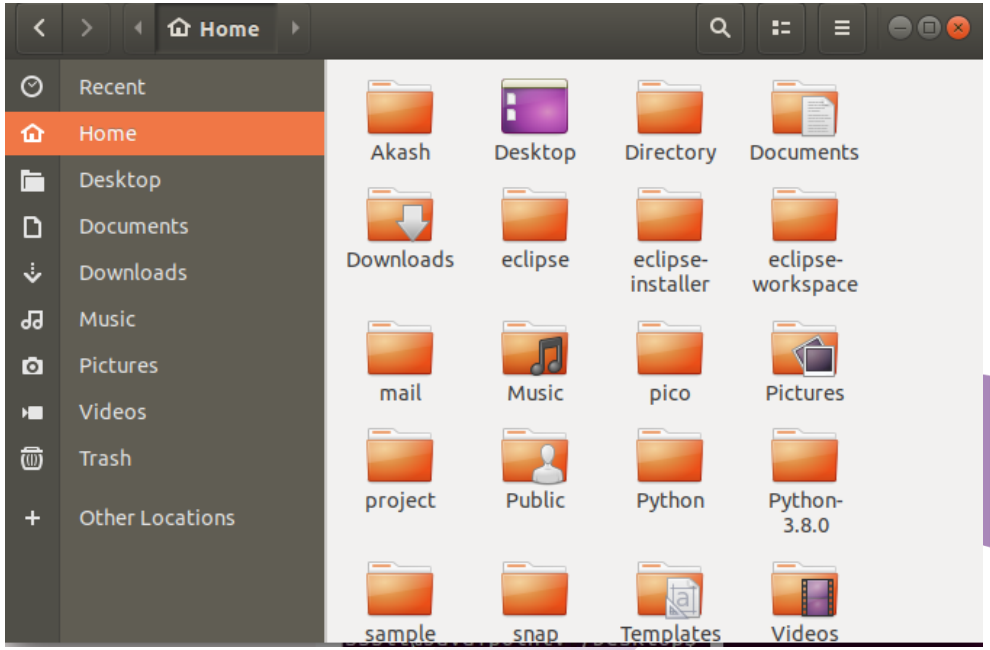


BPP

Logging into a Linux system

What you need to know…

- Two types of login screens
- virtual consoles (text-based)
- graphical logins (display managers)
- Login using login name and password
- Each user has a home directory for personal file storage



Linux home directory

Image source: *JavaPoint*

Shell prompt

What you need to know…

- Once a shell is started
  - Initialisation
  - Waiting for command
- Prompt gives useful information

```
[user@mece_svr  ~]$
```

Logged in User — machine — System connected to a regular user

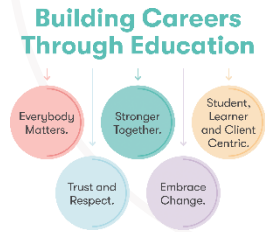
```
[root@mece_svr :~]#
```

Logged in User — System connected to a super user

*This root different from the file system root

BPP

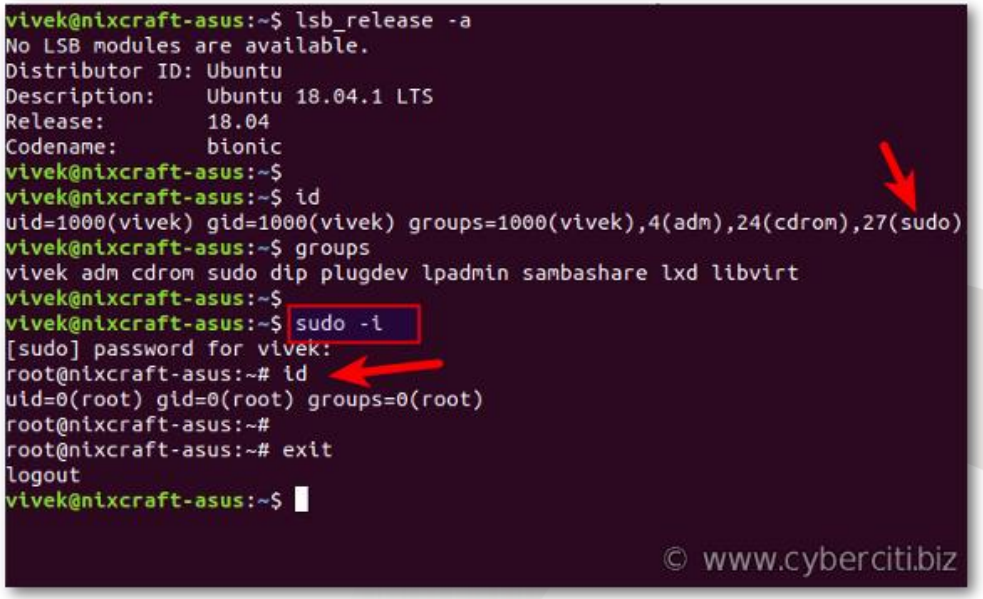
Superuser, root



What you need to know…

- Root is all powerful* (do not leave this account open)
- Similar to administrator account on Windows
- Normal user accounts can only do a subset of the functions root can do.

- ➢ Root access may be required to
- ➢ Install programs
- ➢ Start/ stop an application
- ➢ Change/ access configuration files

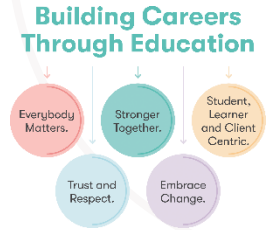


Linux Login as Superuser (root user) command

Image source: *nixCraft*

BPP

Running As Unprivileged User/Group



- every process “runs as” some user
- extremely important this user is not root
  - since any bug can compromise entire system
- may need root privileges, e.g. bind port
  - have root parent perform privileged function
  - but main service from unprivileged child
- user/group used should be dedicated
  - easier to identify source of log messages

BPP

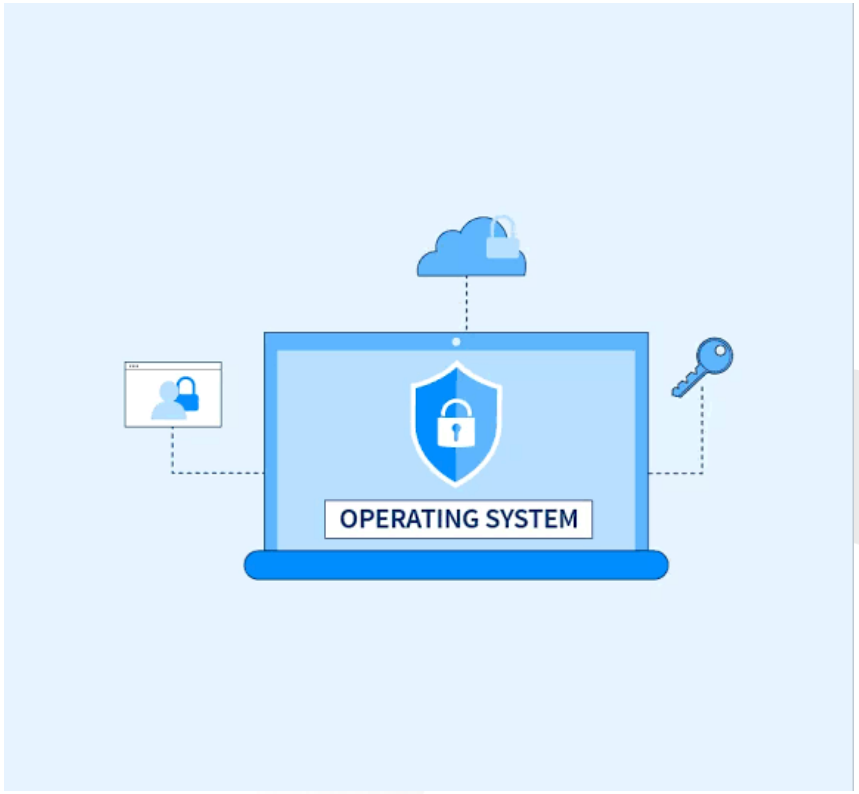
Linux security model

Building Careers Through Education
Everybody Matters.
Stronger Together.
Student, Learner and Client Centric.
Trust and Respect.
Embrace Change.

What you need to know…

Linux's traditional security model is:

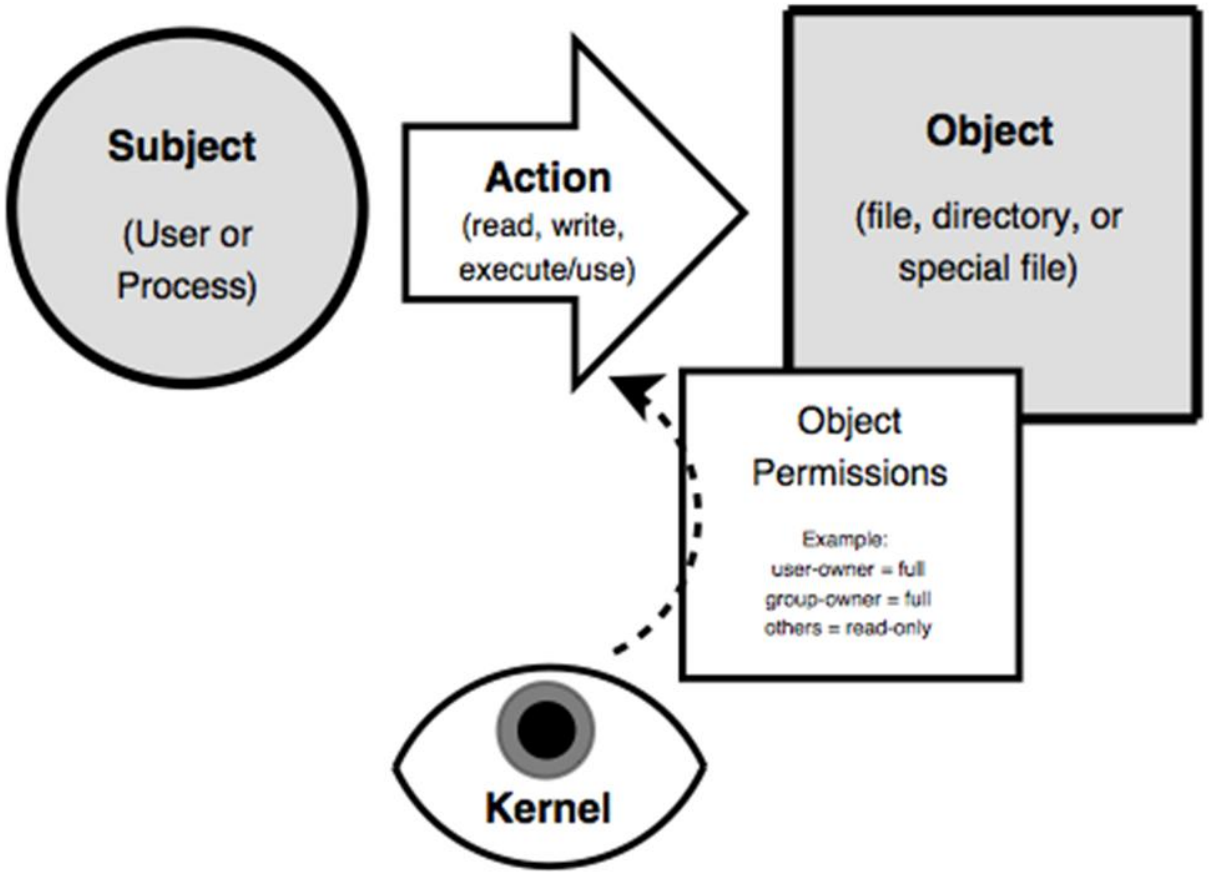
- people or proceses with "root" privileges can do anything
- other accounts can do much less
- hence attacker's want to get root privileges
- can run robust, secure Linux systems
- crux of problem is use of Discretionary Access Controls (DAC)



BPP

Traditional DAC security transactions

What you need to know…

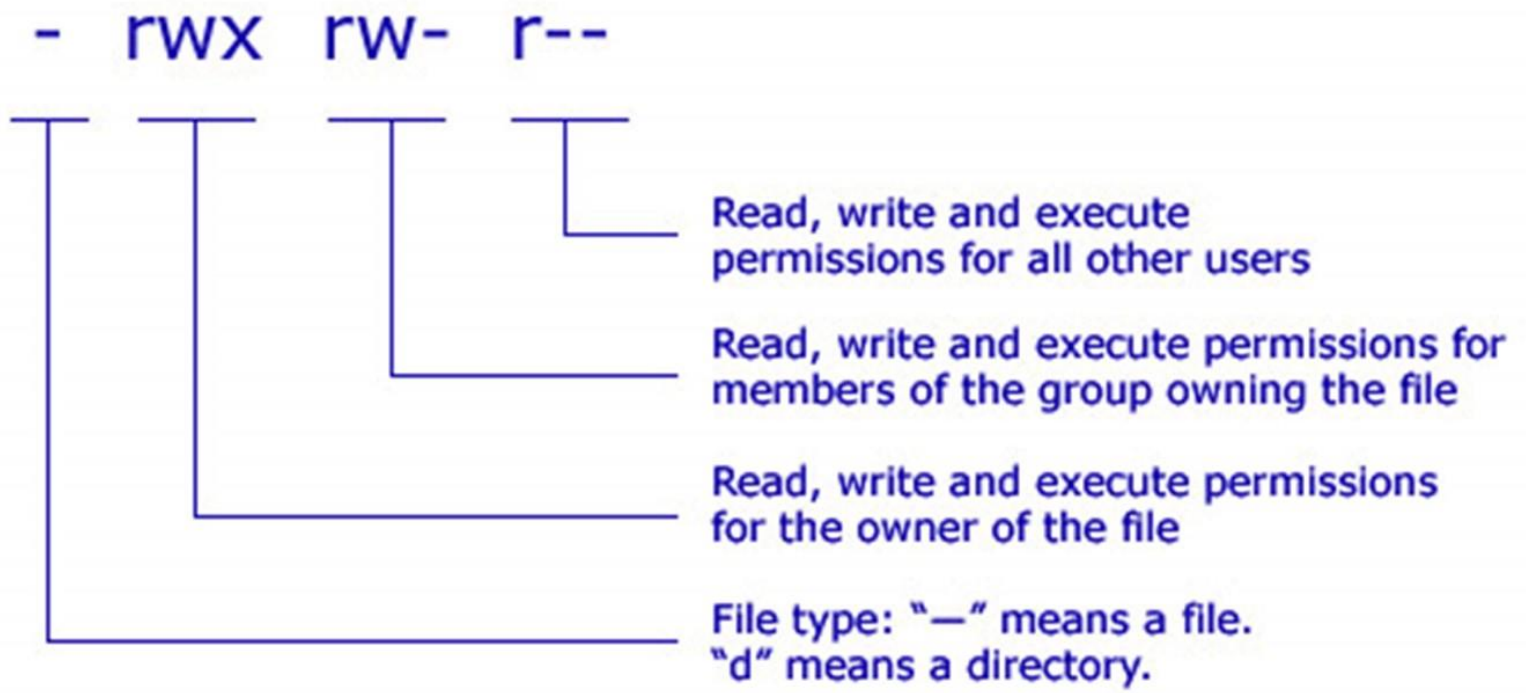


File permissions

What you need to know…

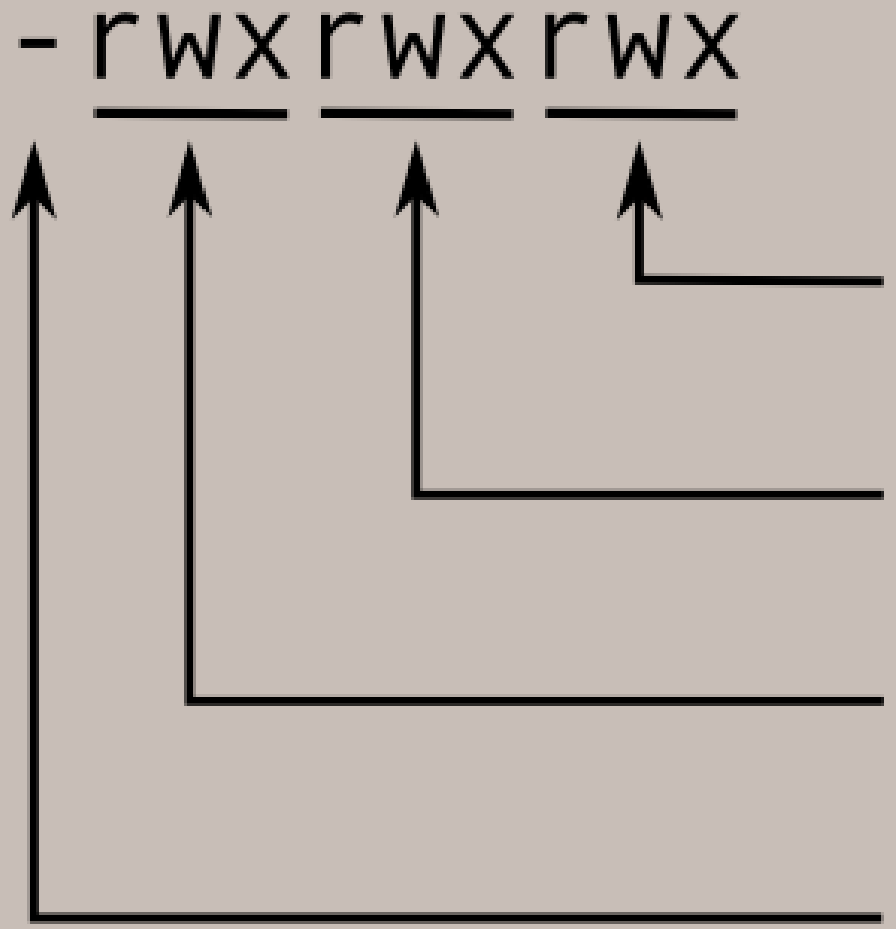
- files have two owners: a user & a group
- each with its own set of permissions
- with a third set of permissions for other
- permissions are to **read/write/execute** and set in this order:

  user group other
- set using **chmod** command



BPP

Building Careers Through Education
Everybody Matters.
Stronger Together.
Student, Learner and Client Centric.
Trust and Respect.
Embrace Change.



Read, write, and execute permissions for all other users.

Read, write, and execute permissions for the group owner of the file.

Read, write, and execute permissions for the file owner.

File type:
- indicates regular file
d indicates directory

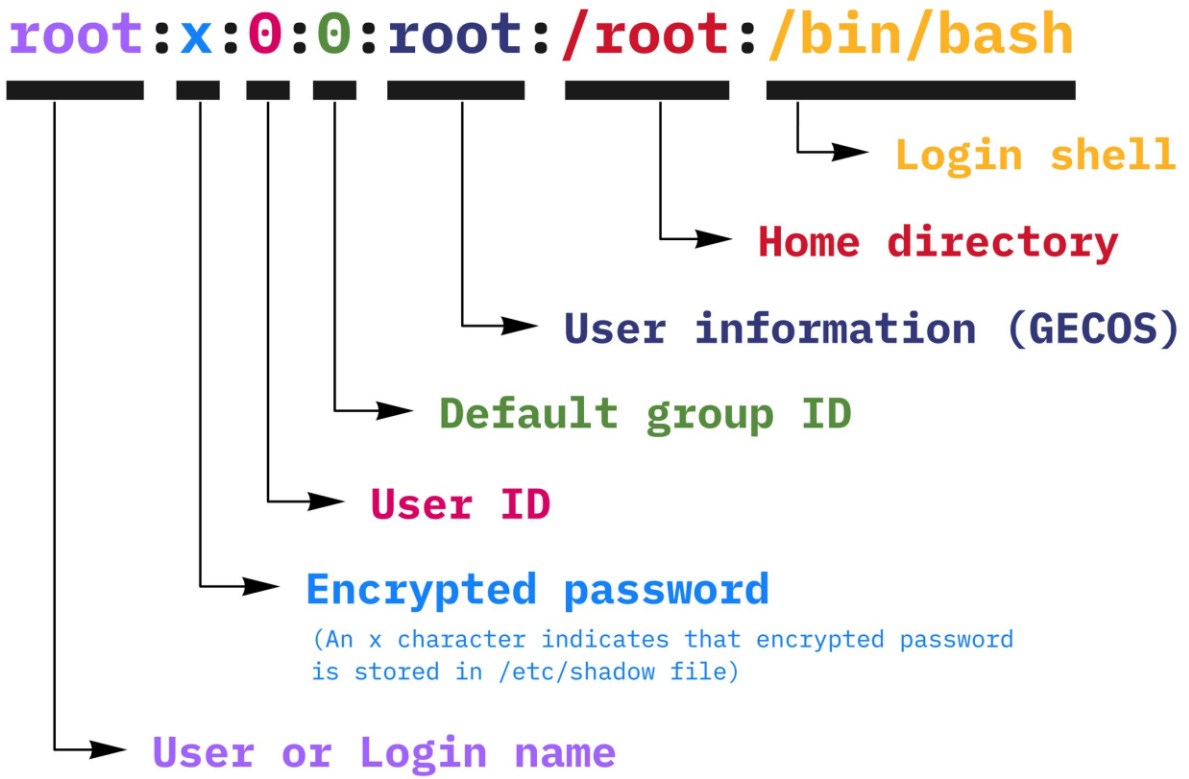
BPP

Users and groups

What you need to know…

- **user's details are kept in /etc/password**
  - See
- **additional group details in /etc/group**

  “pianists:x:102:maestro,volodya”
- use **useradd, usermod, userdel** to alter



A visual display of /etc/passwd and/etc/shadow
Image source: *Statosphere IPS*

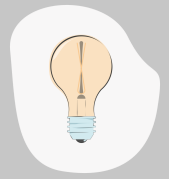
Understanding Linux

Linux commands

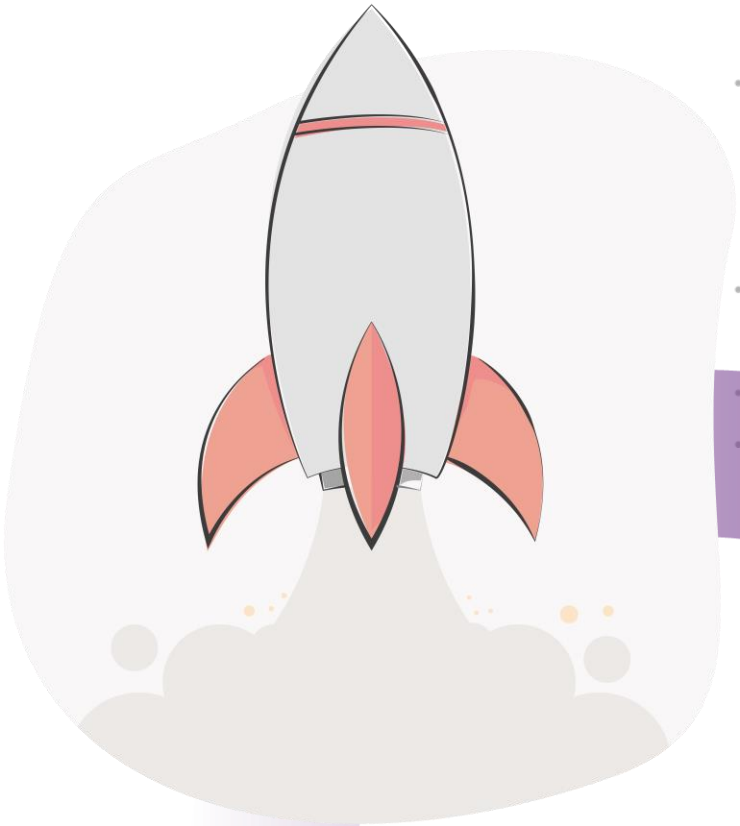
Commands are case sensitive

- Ls ≠ ls ≠ LS

- ls – list directory contents
- cd – change the current directory
- pwd – display the present working directory
- cat – concatenates and display files
- echo – Display arguments to the screen.
- man – Display the online manual
- exit – Exits the shell or your current session
- clear – Clears the screen



Most commands have additional arguments (options) you can add.



Helpful

BPP

Understanding Linux

Linux commands

- [command]--help or [command]-h, gives more info on particular command

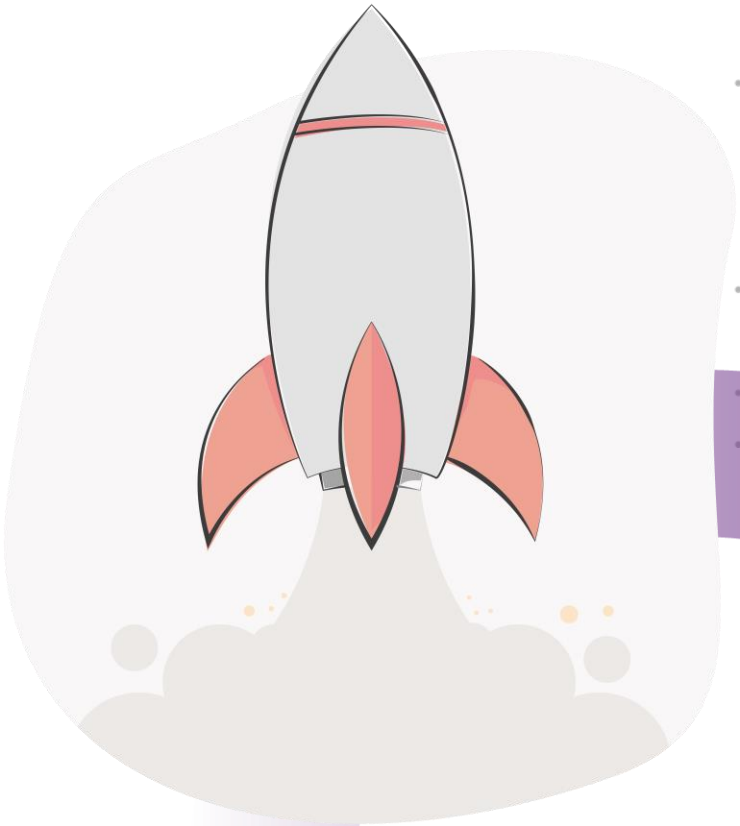
- Sometimes doesn’t work (use man instead then)

- man –k [SEARCH_TERM], for when you are not sure of the command to use)

Basic commands:

- cd, change directory
  - cd ..
  - cd –
  - cd ~/mydir
  - cd /home/usman
  - cd
- su, switch user, su – (complete user environment)

- passwd, change password



Helpful

BPP

Basic Linux commands

Commands and description

Command	Description	Command	Description
cd, change directory	cd .., cd -, cd ~ / mydir, cd /home/usman, cd	cat,concatenate /display files	Cat/home/usman/myfile
su	Switch user, su - (complete user environment)	clear	clears the screen
passwd	change password	date, see/modify system data & time	date, date [MMDDhhmm[[CC]YY][.ss]]
ps, process information	process information: ps, ps aux, pstree, ps fax, top	df, disk space usage	df -h
Free, memory information	Free -m	du, file space usage	Du -sh
Cal, calendar information	Cal, cal 2009	uname, print system info	uname [-a, -s, -n, -r, -v, -m]

*more than one file may be moved at a time if the destination is a directory

BPP

Basic Linux commands (cont)

Commands and description

Command	Description	Command	Description
mv, move and/or rename files and directories	mv [options] *file destination* mv [options] *file1 file2 destination**	**man**, manual pages	man <command>
mv, the destination also works like cp	mkdir, creates directories mkdir <directory name/path> rmdir, removes empty directories rmdir < directory name/path>	**info**, information pages	info <command>
touch, create empty files or update file timestamps	touch <file name/path>	**command --help**, basic help by author	ls --help, man --help
rm, remove files	rm [options] <file name/path> rm -i file (interactive) rm -r directory (recursive) rm -f file (force)	**pwd**	present working directory
rm –r, recursively removes directory trees	rm –rf <directory name/path>		

Basic Linux Commands

- **cat**, concatenate/display files
  - cat /home/usman/myfile
- **clear**, clears the screen
- **date**, see/modify system date & time
  - date, date [MMDDhhmm[[CC]YY][.ss]]
- **df**, disk space usage
  - df –h
- **du**, file space usage
  - du –sh
- **uname**, print system info
  - uname [-a, -s, -n, -r, -v, -m]

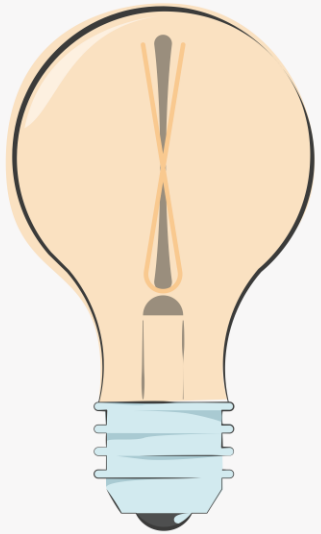
BPP

Shell scripting basics

Commands can be saved in a named file for on-demand (or scheduled) execution

```
#!/bin/sh
MSG=“Assignments due today…”
for i in $(cat ~/users) ; do

    NAME=$i
    EMAIL_ADD=$NAME@ncp.edu.pk
       echo $MSG | mail –s “Warning” EMAIL_ADD
   done
```



Shell scripting

Shell Scripting Basics

```
#!/bin/sh
echo “Welcome to Linux -- ” $USER
echo “Today is : ” $(date)
echo “You are working in : ” `pwd`
echo “Enter your last name: ”
read LNAME
echo "Hello  --  $LNAME”
read X
read Y
echo “The product is : $X*$Y “
echo “Bye Bye…”
```

Getting help on Linux commands

- `[command]--help or [command]-h`, gives more info on particular command
  - Sometimes doesn't work (use `man` instead then)

- `man -k [SEARCH_TERM]`, for when you are not sure of the command to use)

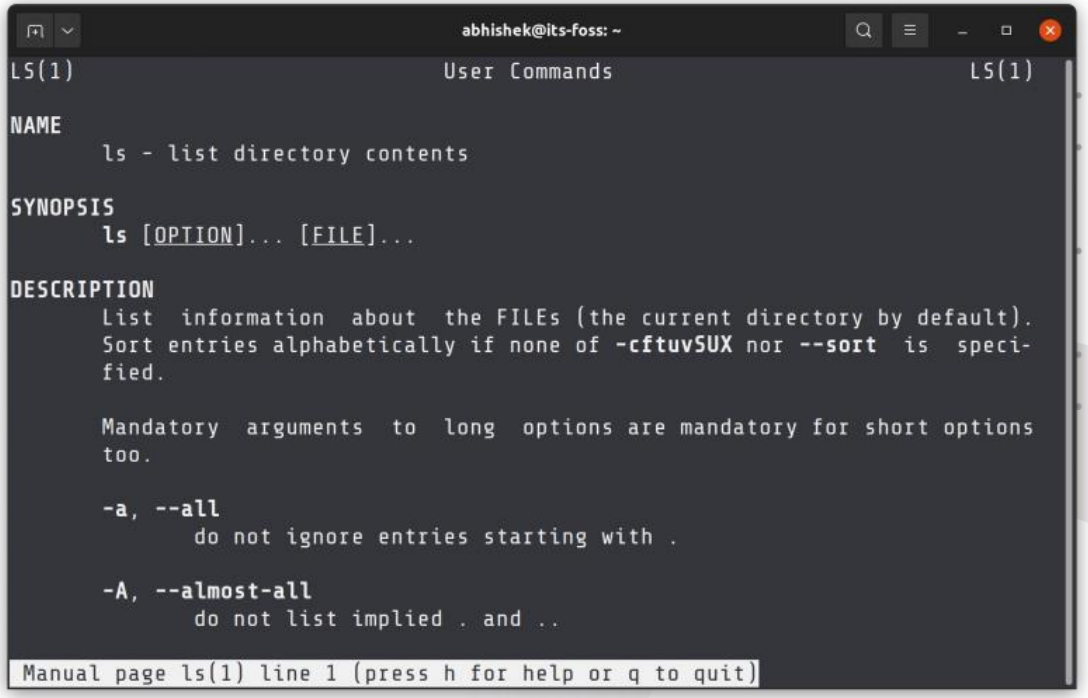
‘man’ – manual of Linux commands

Building Careers Through Education

Everybody Matters. Stronger Together. Student, Learner and Client Centric. Trust and Respect. Embrace Change.

Navigating the ‘man’ pages…

- **Enter –** Moves down one line
- **Space –** Move down one page
- **g –** Move to the top of the page
- **G –** Move to the bottom of the page
- **q –** Quit



An example of a man page in Linux

Image source: *Statosphere IPS*

BPP

Knowledge Check Poll

Which command would you use to remove the directory containing all the files?

a. mkdir

b. ls

c. rm –r

d. rmdir

Submit your responses to the chat!

Knowledge Check Poll

Which command would you use to remove the directory containing all the files?

a. mkdir

b. ls

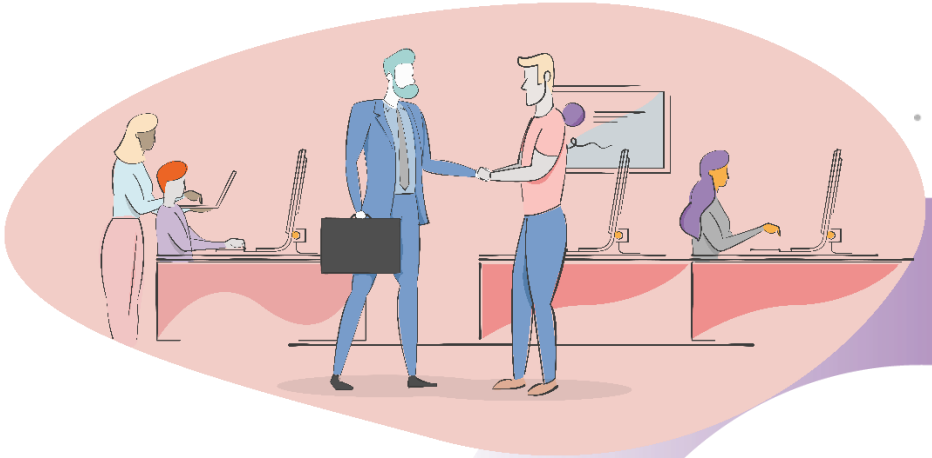
c. rm –r

d. rmdir

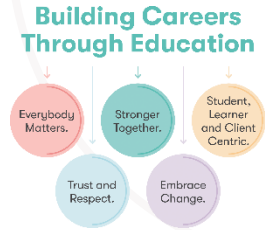
Feedback: c – In Linux, `rmdir` and `rm -r` are both commands that remove directories, but rmdir requires an empty directory to delete.

Submit your responses to the chat!

Practical Linux lab



Creating a system health monitoring script



Group practice

Objective: to create a robust System Health Monitoring Script. This script will serve as a proactive tool, providing valuable insights into your system's health and performance.

We will guide you through four key steps:

- **Writing the Script**: We'll start by opening a text editor and writing the script.
- **Making the Script Executable**: Next, we'll modify the script's permissions to make it executable.
- **Running the Script Manually**: We'll then demonstrate how to manually execute the script.
- **Scheduling the Script**: Finally, we'll add the script to the crontab to ensure it runs daily.

Group practice

BPP

Searching for Files and Directories in Linux

Group practice

Objective: equip you with the essential skills to efficiently navigate and manage your Linux file system.

We will cover the following key areas:

- **Using ‘find’ to Search for Files and Directories**: We’ll demonstrate how to use the ‘find’ command to locate files and directories.
- **Using ‘grep’ to Search Inside Files**: We’ll show you how to use ‘grep’ to find specific content within files.
- **Changing File Permissions**: We’ll guide you through the process of modifying file permissions to enhance security.
- **Using ‘at’ for One-Time Tasks**: We’ll explore how to use the ‘at’ command to schedule one-time tasks.
- **Using ‘cron’ for Regularly Scheduled Tasks**: Finally, we’ll delve into using ‘cron’ for tasks that need to run on a regular schedule.

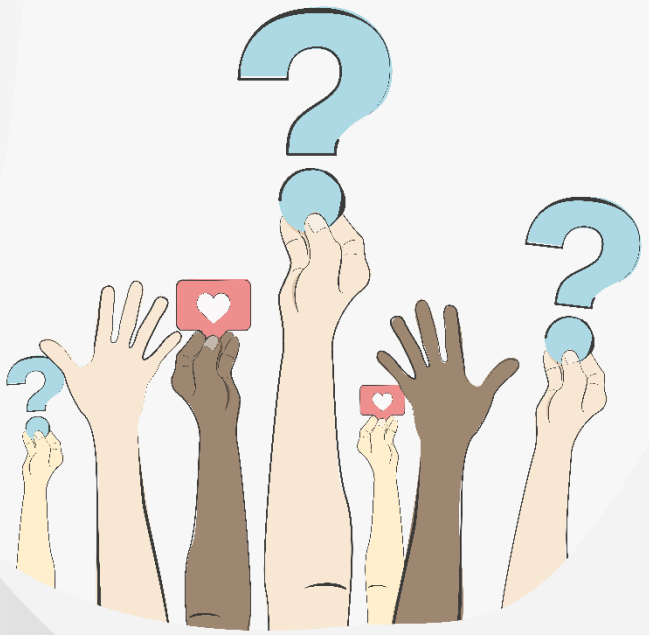
Group practice

BPP

After the webinar

- In your learning journal, write up your research about how Linux is used in your company
  - Which distribution is used
  - How many servers run it
  - What are the use cases
  - What is your colleagues' experience of using it

BPP

Thank you

Do you have any questions, comments, or feedback?