linux Command (Rocky/AWS/Alma) :

* **Super user (admin) :** 
  + To switch back to root user(admin) on AWS linux:
    - **sudo -i** (AWS)
    - **sudo su** (AWS)
      * Ec2-user does not have enough privilege so we need to make it root user by **sudo -i** or **sudo su** (super user do switch user - make me root user)
    - **su root** (**Alma**/ aws with root password)
    - Note: All three commands provide **root-level access and privileges**, but **sudo -i is best** for fully getting the root’s environment and **su root** has to provide the root password.
* **User and information:**
  + To check the current user
    - **whoami**
  + To create user
    - **useradd** Anas or **adduser** Anas
  + To give the user (Anas) the password (as root user)
    - **passwd** Anas
  + To rename user name:
    - **usermod -l** mYnewUser oldAnasUser (warning: do with caution and also if has running process it won't execute so we need to kill the process **pkill -u** oldAnasUser )
  + To check user id & group
    - **id** Anas
  + To substitute user or become different user(operate as diff user):
    - **su** newUser
    - Ex. su Anas
  + To check all users:
    - **cat /etc/passwd**
  + Check specific user:
    - **getent passswd** Anas **(**give the user name here)
  + To delete a user:
    - **userdel** Anas
  + To lockout a user:
    - **usermod -L** Anas (it makes the password invalid)
    - **passwd -l** Anas (It locks the account by specifically altering the password field)
  + To unlock a user:
    - **usermod -U** Anas
  + Force the user to change the password
    - **chage -d 0** Anas
  + To setup the expiration of user account
    - **chage -E 2025-01-01** Anas
    - **usermod -e 2024-12-31** Anas
  + To remove user expiration
    - **chage -E -1** Anas
  + To check when is the expiration for a user
    - **chage -l** Anas
  + To check who is currently log in
    - **who**
* **User group & permission:**
  + To create group: **groupadd** DevOpsTeam
  + To check the group:
    - **getent group**
    - **Cat /etc/group**
  + To add someone to a group:
    - **usermod -aG** DevOpsTeam Anas (include the group then the user)
  + To remove someone from a group called DevOpsTeam:
    - **gpasswd -d** Anas DevOpsTeam (include the user the the group)
  + To add multiple group for a file:
    - **usermod -aG** DevOpsTeam Anas (for first group)
    - **setfacl -m g:qa:rwx** Anas.txt (for 2nd group)
  + To check a specific group by name:
    - **getent group** DevOpsTeam
  + To delete a group:
    - **groupdel** DevOpsTeam
  + Check who belongs to the current group :
    - **getent group** DevOpsTeam (include the groupname : DevOpsTeam)



* + To check group of a user: **id** alice
    - Ex. 
  + To check the file list and folder permission:
    - **ls -ld** ( show the permission, showing long detail, list information about the **directory only, current folder**)
    - **ls -al**l (show all file permission and detail)
    - **ls -ld** file1.txt (finding a specific file permission)
    - **ls -ll** (show non hidden files and folder within current directory)
    - **ls -l** filename.txt(show permission list)
    - **ll** (works like ls but gives list of their permission and users information as well)
  + To change current owner of file or folder:
    - **chown** mike devOpsFile.txt
    - **chown -R** username DirName (flag is used for recursive changes, applying the group change to all files and directories within a specified directory)
  + To allow a group (DevOpsTeam) to have all (RWX) permission for a file only:
    - Step1 : change the file ownership (**group**) :
      * **chown** :DevOpsTeam DevOpsFile (include the group name then filename)
      * **chgrp -R** groupName DirectoryName (flag is used for recursive changes, applying the group change to all files and directories within a specified directory)
    - Step2 : by default the group should have read permission so now to add write and execute   
      We need to make the modification :
      * **chmod g+wx** DevOpsFile (add the file name at the end)



Note: it has given the group full privilege.

* + - Step3 : login a user that is part of DevOpsTeam to see if it got the permission.
  + Allow only read and write permission for group for DevOpsFile:
    - **chmod g=rw** DevOpsFile
  + Allow also write permission for group:
    - **chmod g+w** DevOpsFile
  + Allow only read permission for group:
    - **chmod g=r** DevOpsFile
  + Allow owner to only have read permission:
    - **chmod u=r** DevOpsFile
  + Allow owner to have rwx permission:
    - **chmod u=rwx** DevOpsFile
  + Allow other not to read anything:
    - **chmod o=** DevOpsFile
  + Not to allow any permission for the group:
    - **chmod g=** DevOpsFile (note we are not providing any value to g so change why group has no permission to read, write or execute)
    - It will show 3 desh(- - -) which mean it got no permission to any of them (RWX)
  + Allow everyone rwx:
    - **chmod 777** DevOpsFile
  + Not allow RWX for everyone:
    - **chmod 000** DevOpsFile
  + Not allow anyone not to delete the file except the owner:
    - **chmod +t** fileName
  + To remove sticky bit
    - **chmod -t** fileName
* **User management and session control:**
  + Check who is currently login
    - **Who**
* **Folder and navigation:**
  + To check folder in tree structure : **tree**
  + To go to the root folder will take you to the root : **cd /**
  + To go the folder called tmp : **cd /tmp**
  + To go to the home folder : **cd ~**
  + To check files and folders: **ls**
  + To show the long list : **ls -ll**
  + To show all the files : **ls -al**
  + To Displays information about the directory itself : **ls -ld**
  + To create a folder or directory : 
    - **mkdir** myFolder
      * Ex : **mkdir** AnasFolder
  + To drop or delete an empty folder :
    - **rmdir** foldername
      * Ex: **rmdir** AnasFolder
  + To make nested folders:
    - **mkdir -p** firstFolder/secondFolder/thirdFolder
      * note: **-p** creates parent directories as needed if the folder already exists, give the folder name as parent ex. **mkdir** exitingFolder/secondFolder
  + To drop file or folder with content: **rm -r** foldername
    - **rm -r** helloWorld
      * this will ask permission to delete the file or folder)
    - **rm -rf** helloWorld
      * **-rf** allow to forcefully delete the folder with content inside without asking question)
    - To delete everything from a directory :
      * Erase every folder and files from a directory :
        + **rm -rf \***
      * Delete every file that has letter **F** in them :
        + **rm -rf \*F\***
  + To rename a folder’s name:
    - **mv** oldname newgivenName
  + To go to folder called Anas :
    - **cd /home/Anas**
  + To check the present working directory:
    - **pwd**
  + To go back to previous folder :
    - **cd ..**
      * Ex. **cd ..**
    - To go back to 6 steps back:
      * **Cd ../../../../../../**

****

* + **To move to next folder:** 
    - **cd \*/** (note it only works when next folder has only one folder)
  + To check the current folder size : **df -h**
* **Notifications**
  + To notify everyone:
    - **wall “**hello everyone, please turn off your pc now, we will do work on server**”**
  + To display on green
    - **echo “**hello world**”**
* **Software installation:**
  + To download and install something:
    - **wget** provideTheDownlaobleURLhere
  + Check recent package or software installation history
    - **dnf history**
  + To check if a package called tree installed or not
    - **rpm -q** tree
  + To check if nano install:
    - **which** nano
* **Computer Name and information :**
  + To check hostname/device name :
    - **hostname**
  + To check hostname, os: **uname -a**
  + To change the hostname/computername (root user): **hostnamectl set-hostname pitt**
* **Handy tools:**
  + To clear your screen: **clear**
  + To check history of linux command **: history**
  + To interrupt the command/To get back to command if type cd ` instead of cd~ : **ctrl+ c**
* **To create, read & write files:**
  + To create file : **touch** fileAnas.txt
    - To create single file:
      * **touch** file1.txt
      * Note :
    - To create 10 filesL
      * **touch** testfile**{1..10}.txt**
  + (Nano) To write on a file we can use nano, it’s a text editor for linux : **nano** fileAnas.txt
    - After writing press **CTRL + O** to save then press **Enter**
    - To get out of the Nano, press **CTRL +X**
    - Note : through nano we even create a new file
      * Ex. **nano** newfile.txt
    - To go back to previous state like ctrl + z on nano:
      * Alt + e
  + To read files : cat file1.txt
    - Ex. **cat** anasFile.txt
    - Note: with command cat we can also even write too.
      * ex . **cat >>** newfile (then add your text, it will create the new file called newfile and also add the text for the file once you enter )
  + To go to the root folder : **/**
  + To go to home : **cd ~**
  + To create hidden file
    - **touch .file1**
      * Note: if we add **dot** before file name, by default linux hide those file
      * To check the hidden file we need to **ls -a** means to list all
* **To copy, cut and paste files and folders**
  + To copy file and paste in the same directory :
    - **cp** sourceFile DestinationFile
      * Note : the destinationFile will be newly created, but if there is another file that has the same name then it will ask to override.
      * Ex. **cp** file1 newFile
    - Copy file to a directory:
      * **cp** file1 dir1

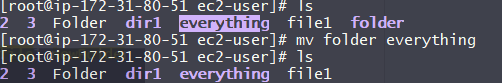


* + - Copy file to somewhere far:
      * **cp** file1 /home/ec2-user/dir1
      * Note: we are in the **current folder/directory path** where the file exists and we are transferring it to dir1 file so we have to give the full path.
  + To copy everything from a folder to another:
    - **cp -r** sourcefolder/\* destinationFolder
    - **cp -r** /home/ec2-user/folder1/\* destinationFolder
  + To cut file:
    - **mv** file1 newFolder



Note: the file and folder needs to be in the same directory or else it will rename instead of cut.

* + To cut or relocate folder:
    - **mv** folder everything



* + To append text from output to a file:
    - **Ip addr** **>>** file1
  + To replace text from output to a file:
    - **Ip addr** **>** file1
* **Search and findings:**
  + To check if there any file named myfile.txt
    - **ls** myfile.txt (from current folder)
    - **find** /home/green/myDir **-name** myfile.txt (if you know the directory)
    - **find . -name** “myfile.txt” (from current folder)
    - **find** **-name** myfile.txt

(it will show all the files with same name from all directory,

note: this needs to execute from root directory)

* + To find a specific text like “hello world” from all files within the /home directory
    - **grep -r** "hello world" /home
    - **grep -ri** "hello world" /home (ignore key sensitivity)
  + Check which file are modified within last 2 days
    - **find** /tmp **-mtime -2**
  + Search anything that start with a and finish with .txt
    - **find** **/** **-name** a**\***.txt
  + Find file that is larger than 100 MG
    - **find / -size +**100M
  + To check file type
    - **file** /etc/passwd
  + To find any user name ec2
    - **grep -ri** “ec2\*” /etc/passwd
* **OS and machine details:** 
  + To check OS: **cat /etc/os-release**
  + Will show you the properties for the system **: hostnamectl**
  + To check ip address:
    - **Ip addr**
    - **hostname -i**
  + To check the linux version:
    - **uname -a**
    - **cat /proc/version**
  + To upgrade the latest patches or update the linux
    - **yum upgrade**
  + To shut down the server
    - **Shutdown -r now**
    - Note : be careful when doing it, only doable off-hour
  + To log out from server:
    - **Exit**
  + Check memory information
    - **free -h** (it shows memory information)
  + Check disk space information
    - **df -h** (it shows disk space information)
  + To check the actual storage given to the OS
    - **lsblk**
* **Process and application:** 
  + To check running process that are there
    - **ps -aux**
    - Note : process is an instance of an application or program that is executed by OS.

Processes can be part of a large application.

An application is like a big toy that can do different things, like play games or show pictures. A process is like one part of the toy working, like when it moves or lights up, while the toy is being played with.

Ex.

Application: Firefox (the program you're using to browse the web)

Process 1: The main Firefox browser window.

Process 2: A separate process running a specific tab.

Process 3: A background process that checks for updates.

* + To check the top 3 most process that are there taking the most cpu consumption:
    - **ps aux --sort=-%cpu | head -n 3**
  + To check the top 3 most process that are there taking the most memory consumption:
    - **ps -aux --sort=-%mem | head -n 3**
  + To find the process from which package
    - **ps -p 123 -o comm=** (provide the parent id or pid, in our case it’s 123)
  + To kill a running process:
    - **kill** 123 (provide the parent id or pid, in our case it’s 123)
  + To check how many applications or packages are installed in os:
    - **dnf list installed | wc -l**
  + Check memory information
    - **free -h** (it shows memory information)
  + Check disk space information
    - **df -h** (it shows disk space information)

ex .

**devtmpfs:** Temporary device files storage like for usb, keyboard etc

**tmpfs:** Memory-based temporary storage used for shared memory (/dev/shm), runtime data (/run), and user-specific runtime data).

**/dev/mapper/...-root**: Logical volume for the root filesystem, housing essential system files.

**/dev/sda1:** Partition for boot files, such as the kernel and bootloader.

* **Drive & partition:**
* **Git & Github:** 
  + Install git

**SSH:**

* To connect with ssh
  + Type **ssh** then the **user@ipaddress**
  + **Ex. ssh Anas@10.0.2.15**
  + Note: the first 3 digits from the left of your local device IP needs to be the same on the running machine(VM).
  + To connect to git bash (AWS):
    - ssh -i “provide the private key.pem” ec2-user@publicDNS
    - Ex . **ssh -i "MainKey.pem" ec2-user@ec2-52-90-114-9.compute-1.amazonaws.com**

**(Recovery)  
linux Command (Rocky/AWS/Alma)** :

**linux Command (Rocky/AWS/Alma)** :

* **Super user (admin) :** 
  + To switch back to root user(admin) on AWS linux:
    - **sudo -i** (AWS)
    - **sudo su** (AWS)
      * Ec2-user does not have enough privilege so we need to make it root user by **sudo -i** or **sudo su** (super user do switch user - make me root user)
    - **su root** (**Alma**/ aws with root password)
    - Note: All three commands provide **root-level access and privileges**, but **sudo -i is best** for fully getting the root’s environment and **su root** has to provide the root password.
* **User and information:**
  + To check the current user: **whoami**
  + To create user: **useradd** Anas or **adduser** Anas
  + To give the user (Anas) the password (as root user): **passwd** Anas
  + To check user id & group: **id** Anas
  + To substitute user or become different user: **su** newUser
    - Ex. su Anas
  + To check all users:
    - **cat /etc/passwd**
  + Check specific user:
    - **getent passswd** Anas **(**give the user name here)
  + To delete a user:
    - **userdel** Anas
* **User group:**
  + To create group: **groupadd** DevOpsTeam
  + To check the group:
    - **getent group**
    - **Cat /etc/group**
  + To add someone to a group:
    - **usermod -aG** DevOpsTeam Anas (include the group then the user)
  + To remove someone from a group called DevOpsTeam:
    - **gpasswd -d** Anas DevOpsTeam (include the user the the group)
  + To check a specific group by name:
    - **getent group** DevOpsTeam
  + To delete a group:
    - **groupdel** DevOpsTeam
  + Check who belongs to the current group :
    - **getent group** DevOpsTeam (include the groupname : DevOpsTeam)



* + To check group of a user: **id** alice
    - Ex. 

* **Files and Folder permission (group level):** 
  + To check the file and folder permission:
    - **ls -ld** ( show the permission, showing long detail, list information about the **directory only, current folder**)
    - **ls -al**l (show all file permission and detail)
    - **ls -ld** file1.txt (finding a specific file permission)
    - **ls -ll** (show non hidden files and folder within current directory)
  + To change current owner of file or folder:
    - **chown** mike devOpsFile.txt
    - **chown -R** username DirName (flag is used for recursive changes, applying the group change to all files and directories within a specified directory)
  + To allow a group (DevOpsTeam) to have all (RWX) permission for a file only:
    - Step1 : change the file ownership (**group**) :
      * **chown** :DevOpsTeam DevOpsFile (include the group name then filename)
      * **chgrp -R** groupName DirectoryName (flag is used for recursive changes, applying the group change to all files and directories within a specified directory)
    - Step2 : by default the group should have read permission so now to add write and execute   
      We need to make the modification :
      * **chmod g+wx** DevOpsFile (add the file name at the end)



Note: it has given the group full privilege.

* + - Step3 : login a user that is part of DevOpsTeam to see if it got the permission.
  + Allow only read and write permission for group for DevOpsFile:
    - **chmod g=rw** DevOpsFile
  + Allow also write permission for group:
    - **chmod g+w** DevOpsFile
  + Allow only read permission for group:
    - **chmod g=r** DevOpsFile
  + Allow owner to only have read permission:
    - **chmod u=r** DevOpsFile
  + Allow owner to have rwx permission:
    - **chmod u=rwx** DevOpsFile
  + Allow other not to read anything:
    - **chmod o=** DevOpsFile
  + Not to allow any permission for the group:
    - **chmod g=** DevOpsFile (note we are not providing any value to g so change why group has no permission to read, write or execute)
    - It will show 3 desh(- - -) which mean it got no permission to any of them (RWX)
  + Allow everyone rwx:
    - **chmod 777** DevOpsFile
  + Not allow RWX for everyone:
    - **chmod 000** DevOpsFile
  + Not allow anyone not to delete the file except the owner:
    - **chmod +t** fileName
* **Folder and navigation:**
  + To check folder in tree structure : **tree**
  + To go to the root folder will take you to the root : **cd /**
  + To go the folder called tmp : **cd /tmp**
  + To go to the home folder : **cd ~**
  + To check files and folders: **ls**
  + To show the long list : **ls -ll**
  + To show all the files : **ls -al**
  + To Displays information about the directory itself : **ls -ld**
  + To create a folder or directory : 
    - **mkdir** myFolder
      * Ex : **mkdir** AnasFolder
  + To drop or delete an empty folder :
    - **rmdir** foldername
      * Ex: **rmdir** AnasFolder
  + To make nested folders:
    - **mkdir -p** firstFolder/secondFolder/thirdFolder
      * note: **-p** creates parent directories as needed if the folder already exists, give the folder name as parent ex. **mkdir** exitingFolder/secondFolder
  + To drop file or folder with content: **rm -r** foldername
    - **rm -r** helloWorld
      * this will ask permission to delete the file or folder)
    - **rm -rf** helloWorld
      * **-rf** allow to forcefully delete the folder with content inside without asking question)
    - To delete everything from a directory :
      * Erase every folder and files from a directory :
        + **rm -rf \***
      * Delete every file that has letter **F** in them :
        + **rm -rf \*F\***
  + To rename a folder’s name:
    - **mv** oldname newgivenName
  + To go to folder called Anas :
    - **cd /home/Anas**
  + To check the present working directory:
    - **pwd**
  + To go back to previous folder :
    - **cd ..**
      * Ex. **cd ..**
    - To go back to 6 steps back:
      * **Cd ../../../../../../**

****

* + **To move to next folder:** 
    - **cd \*/** (note it only works when next folder has only one folder)
  + To check the current folder size : **df -h**
* **OS upgrade :** 
  + To upgrade the latest patches or update the linux**: yum upgrade**
* To download file from net:
  + **wget** provideTheDownlaobleURLhere
* **Computer Name and information :**
  + To check hostname/device name : **hostname**
  + To check hostname, os: **uname -a**
  + To change the hostname/computername (root user): **hostnamectl set-hostname pitt**
* **Handy tools:**
  + To clear your screen: **clear**
  + To check history of linux command **: history**
  + To interrupt the command/To get back to command if type cd ` instead of cd~ : **ctrl+ c**
* **OS and machine details:** 
  + To check OS: **cat /etc/os-release**
  + Will show you the properties for the system **: hostnamectl**
  + To check ip address: **Ip addr**
* **To create, read & write files:**
  + To create file : **touch** fileAnas.txt
    - To create single file:
      * **touch** file1.txt
      * Note :
    - To create 10 filesL
      * **touch** testfile**{1..10}.txt**
  + To write on a file we can use nano, it’s a text editor for linux : **nano** fileAnas.txt
    - After writing press **CTRL + O** to save then press **Enter**
    - To get out of the Nano, press **CTRL +X**
    - Note : through nano we even create a new file
      * Ex. **nano** newfile.txt
  + To read files : cat file1.txt
    - Ex. **cat** anasFile.txt
    - Note: with command cat we can also even write too.
      * ex . **cat >>** newfile (then add your text, it will create the new file called newfile and also add the text for the file once you enter )
  + To go to the root folder : **/**
  + To go to home : **cd ~**
  + To create hidden file
    - **touch .file1**
      * Note: if we add **dot** before file name, by default linux hide those file
      * To check the hidden file we need to **ls -a** means to list all
* **To copy, cut and paste files and folders**
  + To copy file and paste in the same directory :
    - **cp** sourceFile DestinationFile
      * Note : the destinationFile will be newly created, but if there is another file that has the same name then it will ask to override.
      * Ex. **cp** file1 newFile
    - Copy file to a directory:
      * **cp** file1 dir1

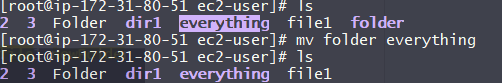


* + - Copy file to somewhere far:
      * **cp** file1 /home/ec2-user/dir1
      * Note: we are in the **current folder/directory path** where the file exists and we are transferring it to dir1 file so we have to give the full path.
  + To copy everything from a folder to another:
    - cp -r sourcefolder/\* destinationFolder
    - cp -r /home/ec2-user/folder1/\* destinationFolder
  + To cut file:
    - **mv** file1 newFolder



Note: the file and folder needs to be in the same directory or else it will rename instead of cut.

* + To cut or relocate folder:
    - **mv** folder everything



* + To append text from output to a file:
    - **Ip addr** **>>** file1
  + To replace text from output to a file:
    - **Ip addr** **>** file1

**SSH:**

* To connect with ssh
  + Type **ssh** then the **user@ipaddress**
  + **Ex. ssh Anas@10.0.2.15**
  + Note: the first 3 digits from the left of your local device IP needs to be the same on the running machine(VM).
  + To connect to git bash (AWS):
    - ssh -i “provide the private key.pem” ec2-user@publicDNS
    - Ex . **ssh -i "MainKey.pem" ec2-user@ec2-52-90-114-9.compute-1.amazonaws.com**