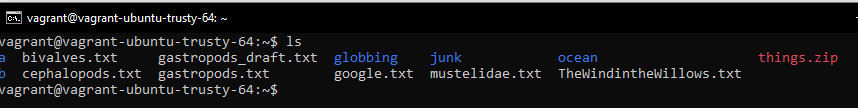
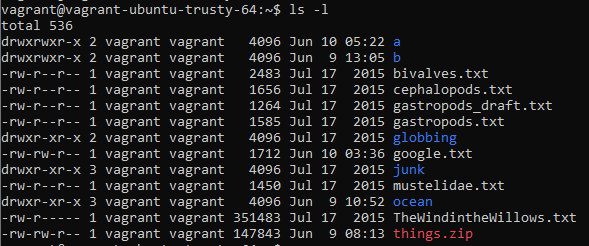
Different Commands in linux: -

1. ls :- List files and directories in the current working directory



We use *-l* to get full details of the files and directories



The first column basically tells about the permissions of the files

It can be broken down :

w xxx yyy zzz

w tells if it is a directory or a file

xxx are permissions for owner

yyy are permissions for group

zzz are permissions for others

xxx, yyy, zzz are if format of *rwx*

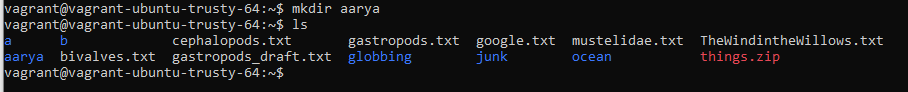
r stands for read permission

w stands for write permission

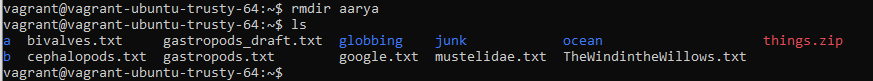
x stands for execute permission

1. mkdir :- Used to create new directories in the current working directory

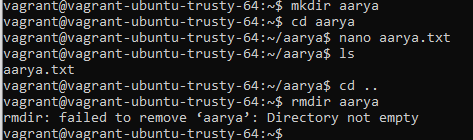
eg:- *mkdir aarya* – This creates a directory/folder named aarya



1. rmdir :- Used to remove the directory or delete an empty directory



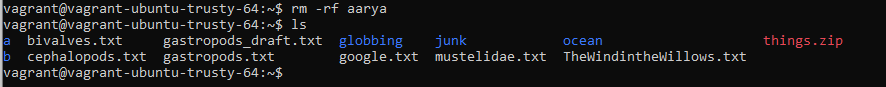
If the directory is not empty then it shows error



1. rm :- Used to remove File or Directory

We use -r to recursively remove the files inside the directory

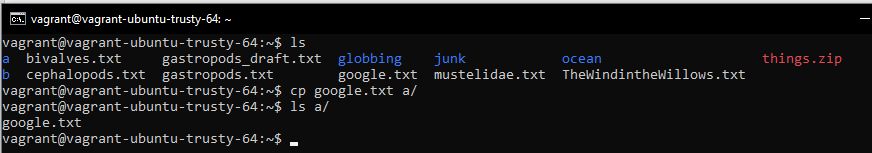
We also use -f to force delete the files



Note: Older version of linux allows the use of “r*m -rf / “* Which is a popular way of hackers crashing your pc. What it essentially does is that it force removes all the files from the root ( root being the top of the directory ).

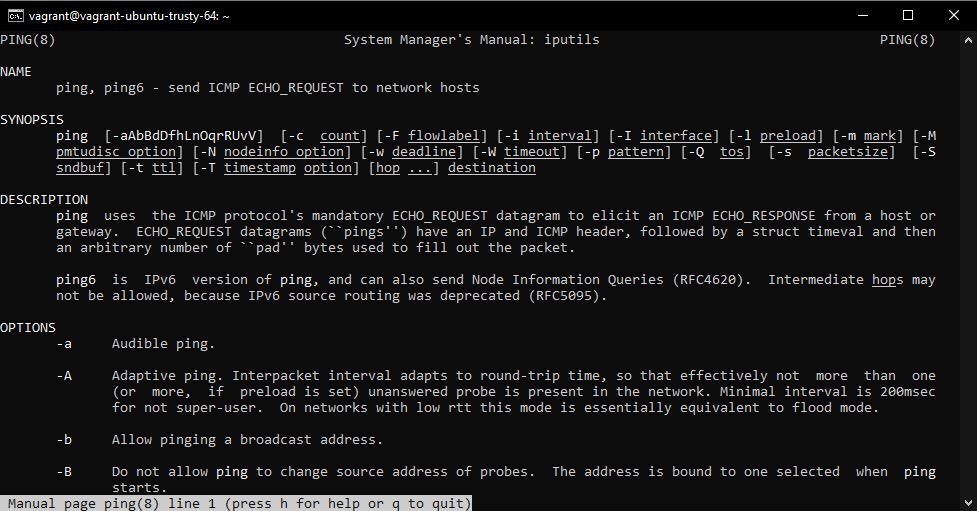
Newer version of linux are safe and show error message if this action was performed

1. cp :- Used to copy file from one location to another location

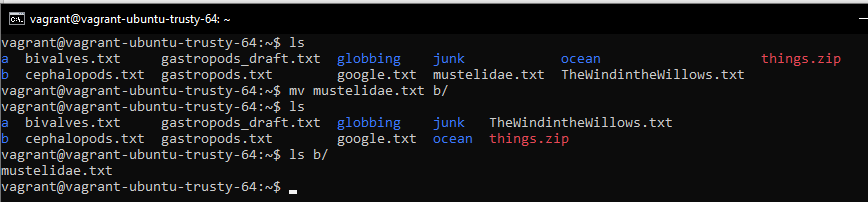


1. man :- Used to see the documentation/manual of a specific command

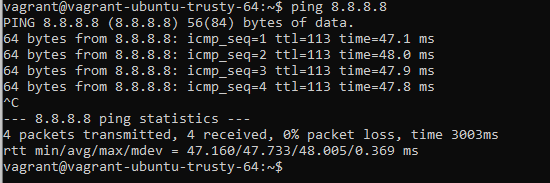




1. mv :- Used to move files from one location to another location



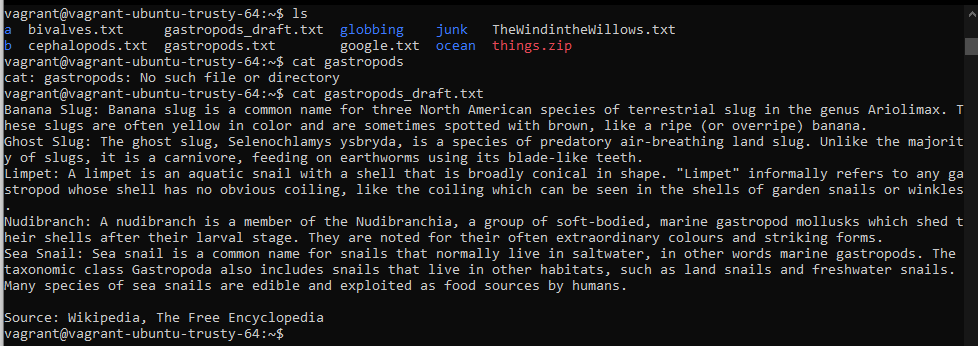
1. ping :- Used to ping a system to see if we have active internet connection



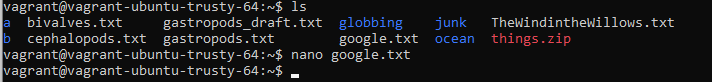
Here, 8.8.8.8 is ip address of google.com.

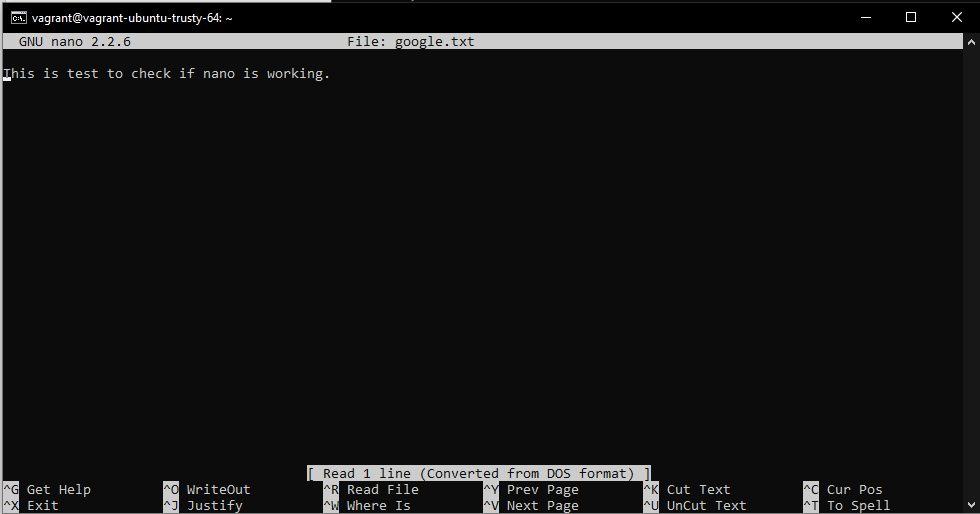
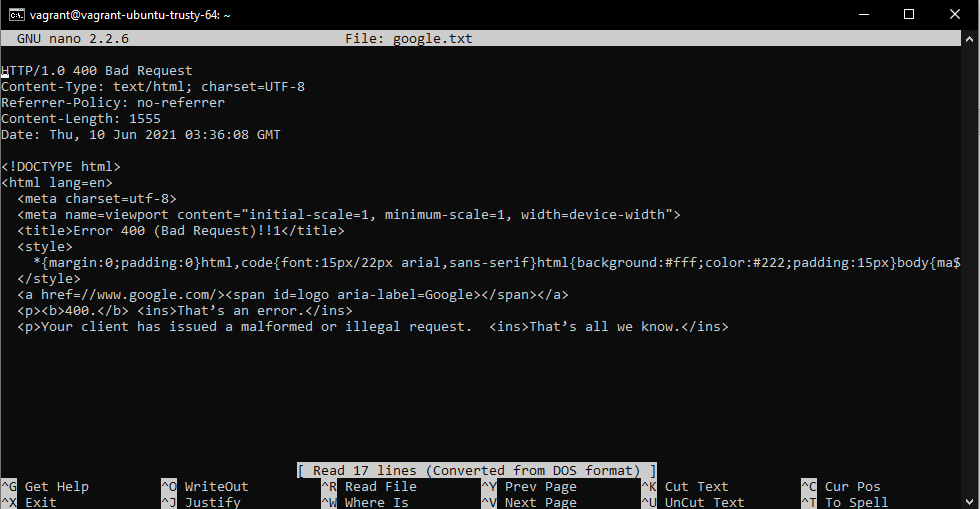
The command runs continuously until an interrupt is given.

1. cat :- Used to read the contents of a file



1. nano :- Used to create/modify a file. It’s command line text editor.

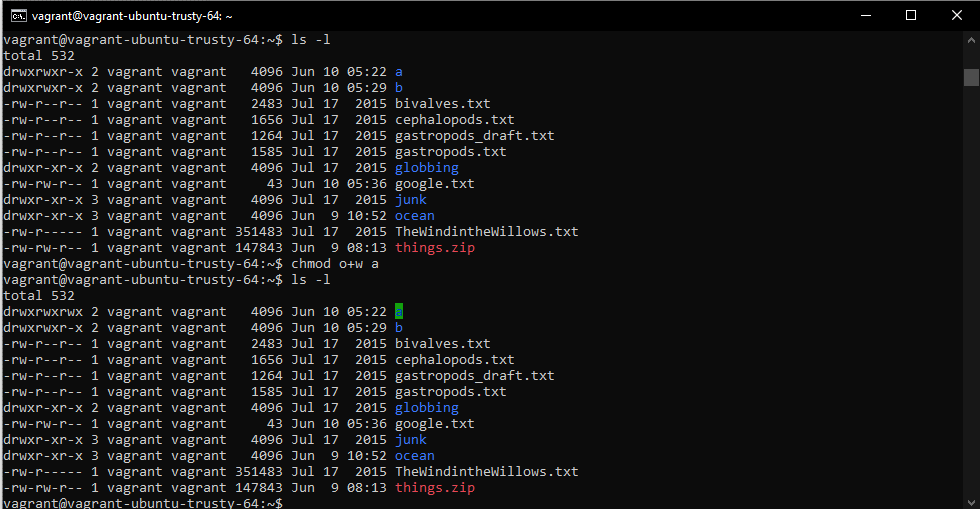






1. chmod:- It is used to change permission on the file or directory

We can use ls -l command to view what are the permissions existing for a file and directory and we can modify accordingly.



Here we used o+w,

o refers to others

+ refers to add

w refers to write

The rest of the semantics which we can use is:-

u -> owner

g -> group

o -> other

+ -> add

- -> remove

= -> set

r -> read permission

w -> write permission

x -> execute permission

We can also use numbers to denote the changes we want

Assume,

r -> 4

w -> 2

x -> 1

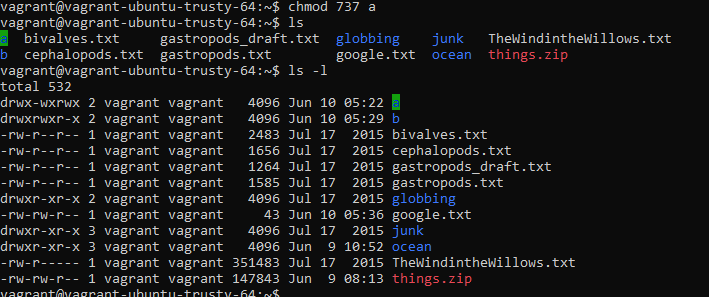
And we want to change the group permission to be only write and execute and owner and others can have all the permissions.

So, we take the sum of the permissions and use it accordingly

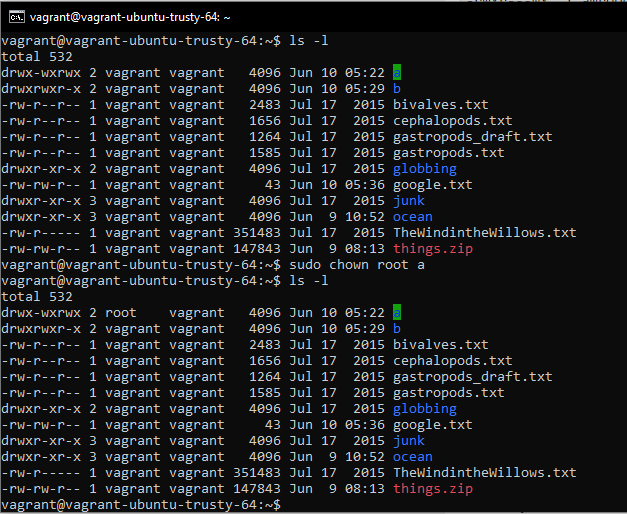
Write and execute = 2 + 1 = 3

Read, write and execute = 4 + 2 + 1 = 7

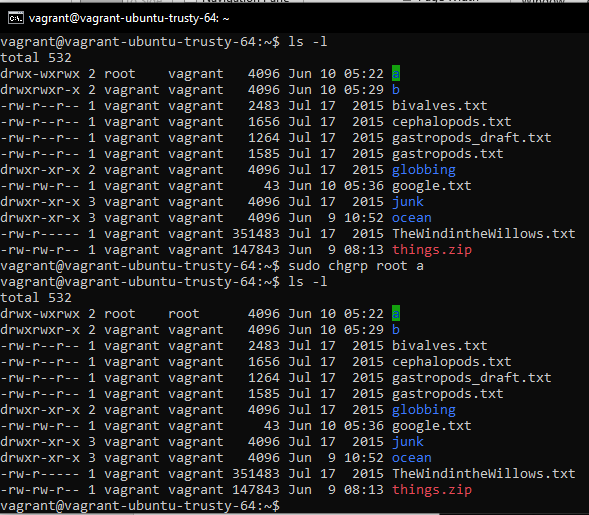
The command we will give is: - *chmod 737 filename*



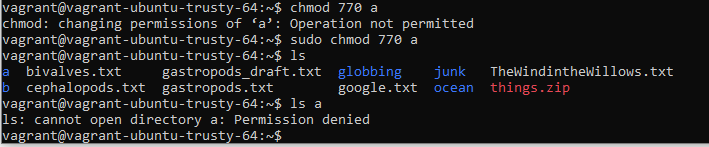
1. chown: - Used to change owner or group for a specific directory or file



1. chgrp: - Used to change the group of specific directory or file



Once the ownership/group is changed and you try to change any permission it wont allow if you are not that particular user.

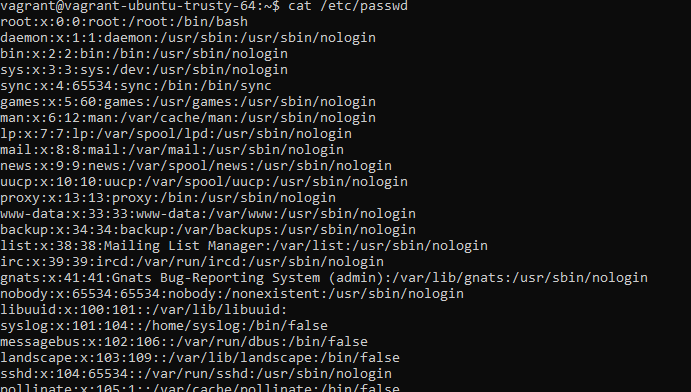


1. awk: - Used for pattern matching and language processing

Normally used with pipes ( | symbol ) so that any input it receives it identifies the pattern and format accordingly

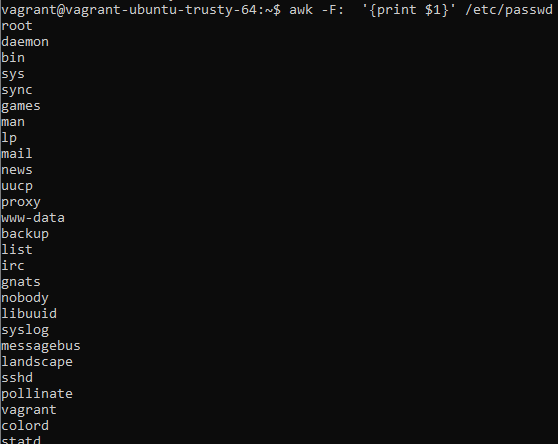
Let’s say we want to list all the users in the system. This is normally found in */etc/passwd* file.

When you print, we see: -



Here we only want to see the first column which is the username

So we use awk to finding pattern and splitting by : and accessing only the first column



Here, -F stands for Field Separator as we are splitting based on :.

Also we use ‘print $1’ to execute what to do for each input and we $1 represents the first word after splitting based on the field separator.

1. Head:- To print first n characters/lines
2. Tail:- To print last n characters/lines