Chatter Permission

```
root@net:~#lsattr
---------------- ./new
root@net:~#chattr +i new
root@net:~#lsattr
----i----------- ./new
```

Small **i** shows that this file is unchangeable and lsattr is a command to check if there is chattr on file.

Before we end up with file permission, let's have little look about numerical file permission.

```
r = 4
w = 2
x = 1
```

The sum of those aforementioned values manipulates the file permission accordingly, that is,

```
root@net:~# ls -al
-rw-r--r-- 1 ali ali 13 Mar 18 07:54 new
```

Here other user only having "read" permission so what we are going to do is to change it into read and write but not execute.

```
root@net:~#chmod 646 new
root@net:~#ls -al
-rw-r--rw- 1 root root 13 Mar 18 07:54 new
```

Let's explore a bit more into it, we want read + write permission so 4 + 2 = 6 that's mean read and write.

Hope it is clear now how to set permission on a file and what it does.

Most Common and Important Commands

ls:	list directory contents
cd:	changes directories
rm:	remove files or directories
chmod:	change file mode bits, from read to write and vise versa
chown:	change ownership of a file
chgrp:	change group ownership
screen:	screen manager with VT100/ANSI terminal emulation, create background process with terminal emulator.
ssh:	secure shell for remote connection
man:	manual/help
pwd:	print name of current/working directory.
cd..:	moves up one directory
mkdir:	create a new directory
rmdir:	remove director
locate:	find a file with in directory or system

whereis:	find a file with in system
cp:	copy file
mv:	move file/directory or rename a file or directory
mount:	mount device such as cdrom/usb
zip:	compress directory/files
umount:	umount(eject) the usb
df:	list partation table
cat:	concatenate the file
ifconfig:	show interface details
w:	Show who is logged on and what they are doing
top:	show system task manager
netstat:	show local or remote established connection
nslookup:	query Internet name servers interactively
dig:	dns utility
touch:	create a file
nano:	file editor
vi:	vim file editor
free -h:	check free memoryruns.

Linux Scheduler (Cron Job)

Cron is a utility that helps us create schedule to perform a certain task/command. As we know that /etc having configuration files for most of the services same as for cron.

We will just go through a quick review of how does it work and how do we set it up.

The following is the hierarchy for it.

```
# * * * * * command to execute
# ┬ ┬ ┬ ┬ ┬
# │ │ │ │ │
# │ │ │ │ │
# │ │ │ │ └───────── day of week (0–6) (0–6 are Sunday to Saturday,
#                                          or use names; 0 is Sunday)
# │ │ │ └────────────── month (1–12)
# │ │ └─────────────────── day of month (1–31)
# │ └──────────────────────── hour (0–23)
# └───────────────────────────── min (0–59)
```

It’s pretty simple and easy to understand; aforementioned hierarchy is self-explanatory.

First * represent min 0-59
Second * represent hour 0-23
Third * represent day of month 1-31
Forth * represent month 1-12
Fifth * represent day of week 0-6