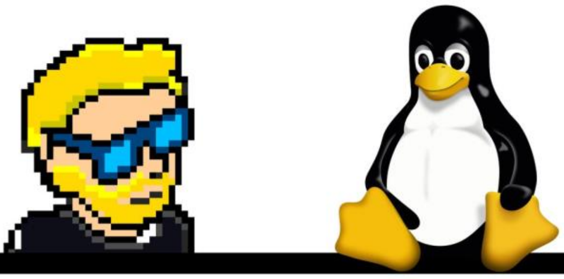
THE LINUX COMMANDS HANDBOOK



- **Short keys in Linux.**
- alt+f1 = to go on application setup
- alt+f10 = maximize window size
- alt+f5 = to reduce window size
- clt+shift+ +++= to increase the font size.
- clt --- = to decrease the font size.
- clt+shift+t = new tab or sub-terminal
- clt+page up = shift from one tab
- clt+page down = shift from one tab
- #alt+f4 ={to close all terminals}
- #alt+f2= {for run bar}
- #clt+c ={ to interrupt the command }
- #clear or #clt+L { to remove all commands}
- #exist or #clt+D {to logout from terminal}

- #shift+page up = {to scroll up the page}
- #shift+page down ={to scroll down the page }
  - ,,,,))))))))))))))

(((((((((os monitoring commands))))))))))

```
        # cat       /etc/system-release
        # cat       /etc/os-release
        #cat      /etc/redhat-release
```

((((((((((((((RAM check commands))))))))))))))

```
# free
# free  -k   {memory in kb}
# free  -m   {memory in mb}
# free  -h    {memory in human readable form}
```

(((((((((((((hard-disk size))))))))))

- #fdisk -l   {partion disk size}
- #du   -sh   {disk utilzation sum in human readable form}
- #df   -h   {this shows mounting points of disk}
- #df   -TH   {with type tells}
- #lsblk

((some important commands))

- #who {Tell who is login}
- #id username {To check id}
- # w   -f   {From }
- # w   -h   { To remove header }
- #whoami   {Tells current login user}
- #uptime   {Machine uptime}
- #pinky   {provides same info}
- # man   touch { Man commands tells command options in brief}
- #useradd   --help   {same as explain above but in shortest manner}

- #touch --help
- #which {which command tells the location of a command where it is stored}

(((((((stat command)))))))

- This command tells full status of "file,,,,,"directory"
- #stat /abc { file status }
- #stat /myfolder {folder status}

(((((((((((file command)))))))))))))

- This command tells us type of content i.e., it is a ""filex or folder"
- # file abc { it will tell us that abc is an file }.
- #mkdir /krishantechblog
- #file /krishanblog { it will tell us that krishantechblog is an folder}.
- #touch /krishantechblog
- #file /krishantechblog

Types of ""command prompt""

- [root@localhost~]#
- #=previllised mode,,,it has 100% rights ..,,,# shows that "root is log in ie.,, super user"
- root= username
- ~= present working directory.
  - o localhost=machine name..

- if i am ''log-in '' with any ''normal user ''
  - o [krishantech@localhost~]$
    - $=unprevilllaised mode ,,,$ shows normal user ,,,it has 10% rights ,which can be increased by root later as needed.
  - o krishantech= username

  - localhost=machine name
    - ~= pwd

HOW to ""add any user''''' & provide "password"

<<<<<How to ''''delete any user''''.

- #useradd krishantech
- #passwd   krishantech {password for krishantech user}
  - 123
  - 123   {password}

- #userdel   krishantech   {by this command only krishantech user will be deleted,,but data of krishantech user will not deleted}.

```
#ls   /home  {in home directory by default users added}
#  rm -rf  /home/lalit   {to delete data of krishantech }
#ls    /home
```

'mkdir,,ls,,cd,,,touch,,,echo,,,,wall,,,,cat,,,,tac,,,wc'

How to make "directories/folders"

mkdir   (is used to make directories or folders)

cd   (is used to change directory : from one directory to another direcotry;)

ls   (is used to for listing of contents)

touch   (is used to make blank files)

echo (is used to print any message:)

wall (is used to print any message over the entire network..)

cat   (is used to   only read the contents of any file..)

wc   (is used to count the words ,,,characters,,,no. of lines in file.)

- **<<<<<<mkdir ,,cd,,,ls,,, command>>>>>>>>>**
- #mkdir   krishan tech blog {mltiple folders   create }
- #mkdir   war   /ret   /world   {multiple folders create}
- #mkdir   /world/yum
- #mkdir   /world/yum/ddd
- #mkdir   /krishantech { Create a directory on / }
- #mkdir   krishanblog
- #ls   /
- #ls

```
#useradd ram
#passwd ram
123
123
#ls   /home
#userdel  -rf    ram {to delete account and data both}
-r= remove
-f = forcefully

#ls  /home

#userdel  --help   {to see options}
```

- **Ls command some examples of ls comman.**
- #ls
- #ls /
- #l.   [for hidden files]
- #ls -a {for all hidden/unhidden]
- #ls -i   {for inode number check of files only]
- #ls   -id   {for inode number check of folder only}
- #ls -l   {to check permissions of files}
- #ls   -ld   {to check permission of directory]
- #ll   {to check permission in long format]
- #ls   -lrt {list   -long format with respect to time}
- #ls   -lrth   {{list   -long format with respect to time in human redable form}.
- #lsof   (list of open files)

- **symbols in linux.**
    - 1)   ;   {semi-colon,, it is used to run more than ''one command at a time'' }.
    - 2)   >>   {concatnation,,,,,,output of two files in one file}..

Examples of ; (semi-colon)

```
#date
#date ; cal ; who       {three commands will fire at a same time}..

#lscpu  ; date  ; cal

#who ; date
```

head & tail

head by defaults shows 10 lines from top of the file ,,,
tail by defaults shows 10 lines from bottom of the file ,,,,

- #head   <options>   <filename>
- #tail   <options>   <filename>

```
#head     /etc/passwd
#head   /etc/group
#tail   /etc/shadow
#tail    /etc/group

#head  -n   10    /etc/group
#head   -n   17     /etc/passwd
#tail  -n   15       /etc/passwd

#head   -n   35   /etc/passwd    |  cat  -n  |  tail -n  5


#head   -n   35   /etc/passwd    |  cat  -n  |  tail -n  5     >
/krishantechblog


#history
#history    |  tail  -n   26

#history   |  cat  -n   |  head   -n  200  |   tail  -n  15   >>
/krishantechblog

#tail   -f    /var/log/secure    {-f  means flow}
```

- ## Grep,,,Egrep,,,Fgrep

grep= it grep only "one pattern"   from   " one file"..

Drawback of   'grep' covered by 'egrep'

Egrep=it grep   " multiple pattern "   from " one file ".
egerp = enhance grep

Fgrep= it grep "any pattern"   from " multile file".

note::::: "Egrep" perfome both tasks of 'grep' and 'fgrep'....

```
#grep    <options >      <pattern>       <file>
```

- -i = ignore k-sensitive
- -o = only a particular word
- -n = those lines in which word present
- -v = reverse match
- -w =   for   exact word
- -A2 = it shows after two lines.

  - -B3 =   it shows before two lines.
  - -C4 = top & bottom
  - -R = to grep over "folder"..
  - ^ = carrot symbol search   begining of line..
  - $ = dollar symbol search end of line.

Runlevels are basically used to manage the ""log-in environments""

- {init} means """initilaize""
- init 0 == means   initialize the machine   to 0 runlevel...

- #init 0 (power-off the machine)

- #init 1   (single user mode   or trouble shooting mode)

- #init 2   (CLI mode   without network)

- #init 3   (CLI mode   with network)

- #init 4   (unused or   blank)

- #init 5   (GUI mode with network)

- #init 6   (reboot)

There are two methods =
a) Temporary = by using   "init command " && """at the time of machine boot up""

b)Permanent =   #vim   /etc/inittab [configuration file of runlevel]
id:5:initdefault
set 3 here at the place of 5 for CLI mode
:wq!
#init 6

NOTE::::   configuration file of   '''runlevel'' is   "" /etc/inittab""""

NOTE::: """"Two modes   in   Rhel-6 Rhel-7"""
a)Rescue mode= to recover any 'grub' & 'kernel' related issues.

b)Emergency mode= to recover   any   'file-system' related issues.

<<<<<<<<<<<<TARGETS in RHEL-7>>>>>>>>>>>>>>>>>

- there are 4 - types of targets in rhel-7..
- targets are managed by "systemd"
- ""pid"" of target is "1"

- graphical.target = full multiuser mode with "CLI &GUI" mode..
  this is equal to "init 5" in rhel-6.

- b)multi-user.target = full multiuser mode with "CLI" mode.

this is equal to "init 3" in rhel -6.

- c)rescue.target = same defination as in rhel -6 {rescue mode}

- d)emergency.target = same defination as in rhel -6 {emergency mode}

```
#init 6   or   #systemctl reboot
               #init 0    or  #systemctl  poweroff
```

How to manage "targets"

```
#systemctl  isolate  multi-user.target   {temporary in cli mode}
#systemctl   isolate    graphical.target   {temproar in gui mode}
```

Here ""isolate"" is used for "tempoary" purpose..

#systemctl set-default multi-user.target {permanent in cli mode ..ie after reboot
machine will be in cli mode..}

#systemctl set-default graphical.target {permanent in gui mode ...ie...after reboot
machine will be in gui mode...}

NOTE::: #cat /etc/systemd/system/default.target (To check)

<<<<<<<<<<<<<<< USER MANAGEMENT >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

# /etc/passwd (this file contains user related information)

fileds= {username:passwd:uid:gid:comment:home:shell}
this file contains " (7) fileds"

#/etc/shadow (this file contains passwd of user)

fields={username:passwd_encrpted_form:passwdpolicy}
this file contains "(9)" fields.
((1)) filled contain "username".
((2)) filled contain "password in encrypted form".
rest of ((7)) filleds contains "passwd policy"....

#/etc/groups (this file contains gropus related information)