

shell prompt

#	Super User
\$	normal user

simple commands

date	display current date and time
cal	display current month calendar
df	displays disk usages
free	display memory usage

Navigation

cd	change the current diretcory
ls	list the directory

cd - : change to previous working directory
 cd ~username : change to home directory

Exploring Filesystem

file	Determine file type
less	view file contents

ls

- l diplay in long format
- h human readable
- a list all files even those hidden,
- d directory
- F option will append an indictor charactor to the end of listed names
- r display result in reverse order
- S sort by file Size
- t modification time

Manipulating Files and Directories

cp	copy the files
mv	move the files from one folder to other
mkdir	creates a directory
rm	remove files and directories

Manipulating Files and Directories (cont)

ln create Hard and symbolic link

* any character
 ? any single character
 [character] any character that is a member of character
 [! character] any character that is not a member of character
 [[:class:]] any character that is a member of class

command Options : cp, mv

i	interactive	same in mv
a	copy files and directories of all attributes, ownership and permission	-
r	recursively copies directories and contents	same in mv
u	copy the files that don't exists	same in mv
v	verbose, informative message	same in mv

Command Substitution

echo \$(ls)

ls -l \$(which cp) new method

ls -l `which` cp back tick, old method

Process

ps	reports snap shot of current process
top	diskplays task
jobs	list active jobs
bg	place a job in the background
fg	place a job in the foreground
kill	send a signal to process
killall	kill process by name

Process commands

ps aux	BSD style ouptut
fg %1	jobs
kill -1 13456	kill process id 13456

SIGNALS: HUP, INT, KILL, TERM, C-ONT, STOP, QUIT, SEGV, TSTP, WINCH

Find command

cmin	match attributes/files/dir modified, n minutes
cnewer	match file/dir whose content or attribute were last modified n*24 ago
ctime n	match file or direct whose content or attribute were modified n*24 hrs ago
empty	match empty files and direct-ories
group name	match files or direc belonging to group name
iname	pattern like the iname test but case insensitive
inum n	match files with inode number
mmin n	match files or dir whose content were modified n minutes ago
mtime n	match files or dir whose content were modified n*24 hours ago
name pattern	match files and dir with specified wildcard pattern
nouser	match files and dir that don't belong to valid user
nogroup	match files and dir that don't belong to valid group



By **SathyaNarayanan**
 (Sathyanarayanan)

cheatography.com/sathyanarayanan/

Published 19th October, 2021.
 Last updated 9th October, 2023.
 Page 1 of 6.

Sponsored by **Readable.com**
 Measure your website readability!
<https://readable.com>

Find command (cont)

perm match files and directories set to mode specified mode

samefile matches files that share the name same inode number as file name

size n match file size of n

type c match file type of c

user match file and directories belong to user

find %test %action %options

find ~

find ~ | wc -l

find ~ -type d | wc -l

b block

c character special device

d directory

f regular file

l symbolic link

Find Logical Operator

and

or

not

find ~ \(-type f -not -perm 0600 \) -or \(-type d -not -perm -700 \)

find -actions

delete

ls

print

quite

find ~ -type f -name '*.BAR] -print

find ~ -type f -and -name '*.BAR -and -print

Userdefined actions

-exec command{}

find ~ -type f -name 'foo*' -exec ls -l '{}' +

find ~ -type f -name 'foo*' -exec ls -l '{}' ';'

Find Options - scope

depth to process a directory files before the dir itself

maxdepth max num of level that find will descend into a directory when performing test and action

mindepth min num of level that find will descend into a directory when performing test and action

mount direct find not to traverse directories that mounter on other filesystem

Tricks

clear clears the screen

history stores the history

script capture all command execution in file a file

history: CTRL + {R, P }

!!number : command history number

!! : last command

!?string : history containing last string

!string : history containing last string

export HISTCONTROL=ignoredups

export HISTSIZE=10000

Working With Commands

type Indicate how command name is interpreted

which display which executable program will be executed

man Display manual page

apropos Display appropriate command

info Display command Info entry

whatis Display brief description of command

Working With Commands (cont)

alias create an alias for command

man page options

- 1 User commands
- 2 Programming interface for system calls
- 3 Programming interface for C Library
- 4 Special files such devies nodes and drivers
- 5 file formats
- 6 Games and screen savers
- 7 misc
- 8 system administrator commands

Redirection

cat concatenates files

sort sort the file

uniq report or omit repeated lines

grep print lines matching pattern

head prints first few lines of file

tail prints las few lines of file

tee reads stdin and send output to stdout and file

wc count number of line, words, and bytes

Redirection Operator

ls -l /usr/bin >file default stdout to file

ls -l /usr/bin 2>file redirects stderr to file

ls -l /usr/bin > ls- redirects stderr & stdout to file

ls -l /usr/bin &> ls- redirects stderr & stdout to file



By **SathyaNarayanan**
(Sathyanarayanan)

cheatography.com/sathyanarayanan/

Published 19th October, 2021.

Last updated 9th October, 2023.

Page 2 of 6.

Sponsored by **Readable.com**

Measure your website readability!

<https://readable.com>

Redirection Operator (cont)

```
ls -l /usr/bin 2> /dev/null
/dev/null      bitbucket
```

Pipelines

```
| pipelines
ls -l /usr/bin | tee reads stdin and writes
tee out | sort  to files and stdout
```

view the world as SHELL viewed

```
echo * expands all files in the path
echo D* expands all files with starting D
$((expression)) Arithmetic expression
echo 75
$((5*2+3))
ls -l which cp backtick instead of expression
```

Braces Expansion

```
echo Front-{A,-B,C}-Back Front-A-Back, Front-B-Back, Front-C-Back
echo {Z..A} Z Y X W V U .....A
mkdir {2009..2011}-0{1..9} creates a directory for 12 months in
{2009..2011}-{1-0..12} 2009 to 2011
```

Environment

```
printenv Print part of all of the environment
set set shell options
export export environment to subsequently executed programs
```

Environment (cont)

```
alias create an alias for command
```

Login & Non Login Shell

Login Shells :

```
/etc/profile : global configuration
~/.bash_profile : personal startup efile
~/.bash_login. : if ~/.bash_profile is not found, bash attempts to read this script
~/.profile : if neither ~/.bash_profile & ~/.bash_login bash reads this file
```

Non Login shells :

```
/etc/bash.bashrc : Global configuration
~/.bashrc : User configuration
```

Quoting

```
echo This is a test; space is stripped
This is a test
echo This is a test
"This is a test"
echo $(echo foo) $(2+2)
$(echo foo)
$((2+2))
echo the $5 is undefined variable so it total is $500.00 like the total is 00.00
```

```
" " : all special character loses its meaning, exception ($, \, `)
' ' : all special character loses its meaning, and no exceptions
```

Searching For Files

```
locate find files by name
find search for files in a dir
xargs build and exec cmd lines from stdin
touch change the file times
```

Searching For Files (cont)

```
stat display file or filesystem status
```

Networking

```
Ping Send ICMP packets
traceroute Print route packets to a network
netstat print network connection, routing table, interface stats
ftp/lftp Internet file transfer program
wget Non Interactive network downloader
ssh OpenSSH SSH Client (remote login program)
scp secure copy
sftp Secure File transfer program
```

OpenSSH package includes two programs that can make use of an SSH encrypted tunnel to copy file across the network.

- a) scp
- b) sftp

SFTP doesn't require ftp server to connect, it needs only SSH running in machines, any machine that has ssh running, we can able to transfer files to server

Archiving and Backup

```
gzip Compress or expand files
bzip2 A block sorting file compressor
tap tape archiving utility
zip Package and compress files
rsync remote file and dir synchronization
unzip Unzip the gzip files
```

compression algorithm: Lossy and Lossless



By **SathyaNarayanan**
(Sathyanarayanan)

cheatography.com/sathyanarayanan/

Published 19th October, 2021.
Last updated 9th October, 2023.
Page 3 of 6.

Sponsored by **Readable.com**
Measure your website readability!
<https://readable.com>

Hacks

Examples:

```
ssh remote-sys 'tar cf - Document' | tar xf -
```

```
find playground -name 'file-A' | tar czf
```

```
playground.tgz -T -
```

```
find playground -name 'file-A' | tar cjf playgr-
```

```
ound.tgz -T -
```

```
rsync -av --delete --rsh=ssh /etc /home
```

```
/usr/local remote-sys:/backup
```

To copy ssh public to remote machine

```
ssh-copy-id -i xxxxx_ext_2023-08-22.pub
```

```
username@hostname
```

- or Hyphen is the stdin and stdout

Text processing examples

```
cat -A $FILE
```

To find any CTRL character introduced

```
sort file1.txt file2.txt
```

sort all files once

```
file3.txt > final_sor-
```

```
ted_list.txt
```

```
ls -l | sort -nr -k 5
```

key field 5th column

```
sort --key=1,1 --key=2n distor.txt
```

key field 1,1 sort and second column sort by numeric

```
sort foo.txt | uniq -c
```

to find repetition

```
cut -f 3 distro.txt
```

cut column 3

```
cut -c 7-10
```

cut character 7 - 10

```
cut -d ':' -f 1 /etc/p-
```

```
assword
```

delimiter :

```
sort -k 3.7n -k 3.1n -k 3.4n distro.txt
```

3 rd field 7 the character, 3rd field 1 character

Text processing examples (cont)

```
paste file1.txt file2.txt > newfile.txt
```

merge two files

```
join file1.txt file2.txt
```

join on common two fields

Text processing

```
cat
```

concatenate files and print stdout

```
sort
```

sort lines of text files

```
uniq
```

report or omite repeated lines

```
cut
```

remove section from each line of files

```
paste
```

merge lines of files

```
join
```

join lines of two files on a common field

```
comm
```

compare two sorted files line by line

```
diff
```

compare files

```
patch
```

apply a diff file to original

```
tr
```

translate

```
sed
```

stream editor

```
aspel
```

interactive spell checker

Permissions

```
id
```

display user identity

```
chmod
```

change's file mode

```
umask
```

set the default file permission

```
su
```

Run a shell as another user

```
sudo
```

Execute command as another user

```
chown
```

changes file's owner

```
chgrp
```

change file groups owner

```
passwd
```

change a user password

Read,Write, Execute

```
- | rwx | rw- | r- | -
```

type | owner | group | world

type

- regular file

d directory

l symbolic link

c character device

b block device

File Mode [Octal]

```
000 ---
```

```
001 --x
```

```
010 -w-
```

```
011 -wx
```

```
100 r--
```

```
101 r-x
```

```
110 rw-
```

```
111 rwx
```

chmod 0600 file [rw- is set for owner]

Mode symbolic notation

```
ugo
```

user, group, others

```
u+x
```

giving exeute permission for user

```
u=rw,o=x
```

giveing user read write and others execute perm

chmod u=rw,o=x file.txt

Packaging Ssystem

```
Debian
```

Debain, ubuntu, xandros, Style Linspire (.dep)

```
RedHat
```

Fedora, CentOS, Redhat style(.rpm) Enterprise, OpenSUSE, Mandriva, PCLinuxOS

Install from Repo

```
debian
```

apt-get install package_name

```
redhat
```

yum install package_name



By **SathyaNarayanan**
(Sathyanarayanan)

cheatography.com/sathyanarayanan/

Published 19th October, 2021.

Last updated 9th October, 2023.

Page 4 of 6.

Sponsored by **Readable.com**

Measure your website readability!

<https://readable.com>

Operation on Package File

Debian :

```
dpkg --install packagefile
apt-get remove package_name
apt-get update
dpkg --list
dpkg --status package_name
apt-cache show package_name (info about package)
dpkg --search file_name (finding which package installed)
```

Redhat :

```
rpm -i packagefile
yum erase package_name
yum update
rpm -u packagefile
rpm -qa. (list)
rpm -q package_name (check package installed)
yum info package_name
rpm -qf file_name (finding which package installed)
```

storage media

```
mount      mount a filesystem
ummount    unmount a file system
fdisk      partition table manipulator
fsck       check and repair filesystem
fdformat   format floppy disk
mkfs       create a file system
dd         write a block oriented data directly to a device
geniso-    create an ISO 9660 image file image
wodim      Write data to optical storage device
```

storage media (cont)

```
md5sum      calculate md5sum

dd if=/dev/cdrom of=ubuntu.iso (create image file)
fsck /dev/sdb1 (check file system)
mkfs -t ext3 /dev/sdb1 (create ext3 file system)
umount /dev/sdb1 (unmount the file system, before changing the partition table)
fdisk /dev/sdb (create a partition table )
Generate 1GB file using uurandom
dd if=/dev/urandom of=sample.txt bs=64M count=16
```

POSIX charcter class

character class
 alnum, word , alpha blank, cntrl, digit, graph,lowe, punct, print, space, upper, xdigit
 format: [:alnum:]

```
echo $LANG
export LANG=POSIX
```

grep Options

```
i  ignore case
v  invert match
c  print number of matches
l  print the name of each file that contain matches
L  print only the names of files that don't match
n  print match line with the number
h  for multiple output supress the output of filename
```

```
grep [options] regex [file....]
```

metacharacters : ^ \$. [] { } - ? * + () | \

grep examples

```
grep -h '.zip' file.list  . is any character
grep -h '^zip' file.list  starts with zip
grep -h 'zip$' file.list  ends with zip
grep -h '^zip$' file.list  containing only zip
grep -h '[^bz]zip' file.list  not containing b and z
grep -h '^[A-Za-z0-9]' file.list  file containing any valid names
```

grep Quantifiers

```
?  match element zero or one time
*  match an element zero or more times
+  Match an element one or more times
{}  match an element specific number of times
```

It always match the preceding element
 {n} match the preceding element occurs n times

{n,m} match the preceding element occurs at least n times, but no more than m times
 {n,} match the preceding element if it occurs n or more times
 {,m} match the preceding element if it occurs no more than m times

grep hacks

```
grep -E '^\([0-9]{3}\)\.[0-9]{3}\.[0-9]{3}$' phone.txt
locate --regex 'bin/(bz|gz|zip)'
```

```
for i in {1..10}; do echo "(${RANDOM:0:3})-${RANDOM:0:3}-${RANDOM:0:3}"; done >>phone.txt
```



By **SathyaNarayanan**
 (Sathyanarayanan)

cheatography.com/sathyanarayanan/

Published 19th October, 2021.
 Last updated 9th October, 2023.
 Page 5 of 6.

Sponsored by **Readable.com**
 Measure your website readability!
<https://readable.com>

Text Processing Hacks

<code>diff -c</code>	context format
<code>diff -u</code>	unified format
<code>diff -Naur oldfile newfile > diff_file</code>	write the diff to diff file
<code>patch < diff_file</code>	applied to oldfile directly.
<code>echo "secret text" tr a- zA-Z n-za-mN-ZA-M</code>	frperg grkg ROT13 Encoding
<code>echo "frperg grkg" tr a- zA-Z n-za-mN-ZA-M</code>	secret text ROT13 Decoding

-, +, ! : deleted, added, line changes.

```
$diff -Naur file1.txt file2.txt > diff_file.txt
$patch < diff_file.txt
patching file file1.txt
```

rotate by 13 places (ROT13)



By **SathyaNarayanan**
(Sathyanarayanan)

cheatography.com/sathyanarayanan/

Published 19th October, 2021.
Last updated 9th October, 2023.
Page 6 of 6.

Sponsored by **Readable.com**
Measure your website readability!
<https://readable.com>