The Ultimate Linux Cheatsheet

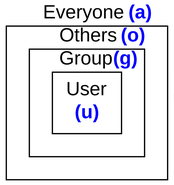
File Operations, Terminal Hotkeys, Utilities, Filters using Regular Expressions, Sed & Awk, VI-Editor, Shell Programming, Processes, System & User Info, Networking, System Administration

File Operations

- Display file contents : **cat <filename>**
- Paginate & print file contents : **less <filename>** (*spacebar* scrolls pages, *q* to quit from read mode)
- File type : **file <filename>**
- Print with line numbers : **nl <filename>**
- Copy file : **cp <file1> <file2>**
- Move/Rename file : **mv <file1> <file2>**
- Make directory : **mkdir <directory>**
- Change directory : **cd <dir>, cd $HOME**
- Display present working directory : **pwd**
- Delete file/directory : **rm <file>, rm -rf <file/directory>** (-f=force, -r=recursive)
- Line/Word/Char count in file : **wc <filename>**

File Permissions

- File access to user based on two parameters: user access level, read/write/execute permission



Read	Write	Execute	None
4	2	1	0

chmod u=rwx,g=rw,o=r <filename

⇕ **Equivalent**

chmod 764 <filename>

- Change user and group ownership : **chown** *user:group* **<filename>**, Change group ownership only : **chgrp** *group* **<filename>**
- Create soft symbolic link : **ln -s <source_file> <destination_name>** (remove -s for hard-link)

Terminal Hotkeys

- Move cursor to beginning of line : **Ctrl+A**
- Move cursor to end of line : **Ctrl+E**
- Move cursor backward by a character : **Ctrl+B**
- Move cursor forward by a character : **Ctrl+F**
- Delete last word typed in : **Ctrl+W**
- Erase complete line : **Ctrl+U**
- Execute last line in shell : **!!**
- Display last executed command : **Ctrl+P**
- Cancel currently running command : **Ctrl+C**
- Close current terminal window : **Ctrl+D**
- Goto home directory : **cd**
- List contents of current directory : **<Tab>**
- Run application window : **Alt+F2**

Basic Utilities

- Location of command : **which <cmd_name>**
- Locate path : **locate <cmd/file_name>**
- Documentation : **man <prog_name>**
- Summary of command : **whatis <cmd_name>**
- Display message : **echo "text to print"**
- Calculator : **bc, quit** to exit
- Record all commands issued in terminal : **script, quit** content stored in filename=*typescript*
- Changing your password : **passwd**
- Pipe (|) redirects output of *utility 1* to *utility 2*. Ex: **cat file | grep hello | wc -l** prints content of file, counts number of lines with the word *hello*
- New pkg install (Mint) : **sudo apt-get install** *pkg*

Regexp's & grep

Symbol	Matches
a*	Nothing or a,aa,aaa ...
.	A single character
.*	Nothing or any number of characters
[abc]	A single character which **is** a or b or c
[^abc]	A single character which is **not** a or b or c
[1-3]	A single digit between 1 to 3
^pat	Pattern *pat* at beginning of line
pat$	Pattern *pat* at end of line

- Search for pattern *pat* in file : **grep "*pat*" <file>**
- Print lines **without** *pat* : **grep -v "*pat*" <file>**
- Ignore case & search : **grep -i "*pat*" <file>**
- List files with pattern in **.** : **grep -l "*pat*" ./***

Sed & Awk

- Stream editing : **sed '<cmds>' file**

Ex 1: **sed -n '1,5p' filename.txt**
Print lines 1-5 ← p=print, i=insert, w=write, q=quit

Ex 2: **sed -n '1,5w nfile.txt' filename.txt**
Quiet mode ← Writes lines 1 to 5 to nfile.txt

Ex 3: **sed 's/pat1/pat2/' filename.txt**
Substitute *pat1* with *pat2* in file

- Search *pat* & execute : **awk '/pat/ <cmds>' file**

Assume **file.txt** has only one line - *Linux is awesome*

Ex : **awk '/Linux/ { print "GNU/"$0"!" }' file.txt**
pattern → execute cmd; original line → append GNU

Output :*"GNU/Linux is awesome!"* printed to screen

The VI-Editor

- Open file in read mode (default) : **vi <filename>**
- Editing modes : **a** (append), **i** (insert), **x** (delete), **Esc** (read), **yy** (copy line), **yw** (word), **p**(paste)
- Delete line(s), word(s) : **dd , 10dd** *for 10 lines*, **dw** *for 1 word*, **d10w** *for 10 words*
- Search & replace : **:[range]s/old_text/new_text/**

:%s/old_pat/new_pat/ → in entire file, substitute
:.s/old_pat/new_pat/ → in current line
:1,10s/old_pat/new_pat/ → in lines 1-10, substitute
:.+10s/old_pat/new_pat/ → in current and next 10 lines

- File management : **:w <file>** (save file), **:wq** (write & quit), **:q!** (quit w/o saving)
- Cursor movement : **h**(←), **j**(↓), **k**(↑), **l**(→)

Shell Programming

- Execute shell script : **sh <script_file>**
- Ex: Script ↓ to compare numbers

```
compare() {                          → Function declaration
if test $1 -gt $2 ; then             → "If" begins
   echo "$1 greater than $2"
fi }                                 → "If" ends
compare 3 1                          → Prints "3 greater than 1"
compare 1 3                          → Nothing
```

- File redirection : > **file** (write to file), >> **file** (append to file), < **file** (read from file)
- Logical Operations : && (and), || (or), ! (not)
- Numeric Tests : **lt** (less than), **gt** (greater than), **eq** (equal to), **ne** (not equal to), **ge** (greater than or equal to), **le** (less than or equal to)

Processes (List, Schedule)

- Table of active processes(programs) : **top**
- Paginated *top* : **ps -axfww | less -S**
- Process ID of current terminal : **echo $$**
- Kill process : **kill <process_id> or xkill**
- Background a job : **<Ctrl-Z>** *followed by* **bg**
- Scheduling job (execute once) : **at -f <script.sh> <time>**, example time=13:00 (or) 1:00pm (or) 1:00pm tomorrow (or) 1:00pm March 8,2013
- Periodic execution : **crontab -e cron.txt**, cron.txt contains: mins(0-59) hrs(0-23) day_of_month(1-31) month(1-12) day_of_wk(0-6) job_cmd.
- Sample cron.txt for an alarm at 7:00 am from Mon to Fri, yr-round : **00 7 * * 1-5 play_alarm**

System & User Information

- System name : **hostname**
- CPU Info : **grep "model name" /proc/cpuinfo**
- RAM Size & Usage : **free -m**
- HDD Size & Usage : **df -h**
- Operating system description : **lsb_release -a**
- Current system date and time : **date**
- Calendar for 3rd month, 2013 : **cal 3 2013**
- Configured hardware : **cat /proc/devices**
- List all devices and properties : **lshal**
- Detected hardware and boot messages : **dmesg**
- Current/last logged in users : **who, last**
- Active user ID : **id (or) whoami (or) logname**
- History of commands issued : **history**

Networking

- Display wireless settings : **iwconfig**
- Current config. of n/w interfaces : **ifconfig**
- Test n/w connection : **ping www.<site>.com**
- Find IP address : **nslookup www.<site>.com**
- Trace IP routing : **traceroute www.<site>.com**
- SSH to a remote machine : **ssh user@ipaddr** (use **-X** to enable X-forwarding of applications)
- FTP : **ftp <host/ipaddr>** (use **get** (to download), **put** (to upload), **bye** (to end session)
- Remote file(s) copy/sync between two machines : **rsync -avrop -aessh user1@ip1:src_path user2@ipaddr2:dest_path** (or) **scp -r u1@ip1:src u2@ip2:dest** (**rsync** is *faster* than **scp**)

System Administration

- Add user : **adduser <uname> (needs su access)**
- Remove user : **rmuser <uname>**
- Directory size listing : **du -sh ***
- List all kernel modules : **lsmod**
- Kernel version information : **uname -a**
- Kernel compilation from source code :

```
cd /usr/src/linux-<ver>   → cd to kernel source location
make mrproper             → Start with clean config files
make oldconfig            → Get missing config options
make menuconfig           → Feature selection
make                      → Create compressed kernel img
make modules              → Compile modules
make modules_install      → Install modules
make install              → Install kernel
reboot                    → Reboot
```