LINUX CHEAT-SHEET

Neel Soni

Folders / Directory	
Create single folder	mkdir <folder name>
Create multiple folders	mkdir <folder name> <folder name>
Create nested folders	mkdir -p <folder 1>/<folder 2>/<folder 3>
Create multiple numbers of folders at once	mkdir <folder name>{starting-range..ending-range} ex. mkdir day-{1..5}
Change directory	cd <folder name> cd ~[username] goes to another user's home directory. cd .. moves one directory up. cd- moves to your previous directory.
Check present working directory (current working folder)	pwd
Files	
Create single file	touch <file name> cat > <file name>
Create multiple files	touch <file name> <file name>
Create nested files	touch -p <file 1>/<file 2>/<file 3>
Create multiple numbers of files at once	touch <file name>{starting-range..ending-range} ex. touch day-{1..5}
View what's in the file	cat <file name>
Merge 2 files and store data in one	cat filename1 filename2 > filename3 Merges filename1.txt and filename2.txt and stores the output in filename3.txt.
Show only top lines from the file	head –[number] [filename]
Show only bottom lines from the file	tail –[number] [filename]
Word Count	wc [options] [filename] -w count number of words -l count number of lines
Find the difference between two files	diff [options] <file 1> <file 2> -c displays the difference between two files in a context form. -u displays the output without redundant information. -i makes the diff command case insensitive

List out Files & Folders	
List out folders and files	ls
List out all the files and folders including hidden with file permissions	ls -la
List out all the files and folders with file permissions	ls -ltr
Lists all the files in the subdirectories	ls -R
Print on screen	
Print	echo <options> “<to be print>” use -e for (\n) new line
Print	printf “<to be print>”
Date and Time	
Current date and time	date
Clear Screen	
Clear the screen	clear
Make current line at the top of the screen without removing data	ctrl + L (windows) cmd + L (mac)
Copy and Paste	
Copy file and paste	cp <option> <source location> <destination location>
Copy all data starting with specific word/character	cp <option> <source-location>* <destination location>
Copy folder having data	cp -r <source location> <destination location>
Delete	
Remove the file and directory	rm <option> <source location> <destination location>
	<options> -r for recursive (used for folders) -v verbose -f for forcefully
Move and Rename	
Move the file or folder	mv <source location> <destination location>
Rename the file or folder	mv <old name > <new name>
History	

Commands you have run till now	history [options] -c clears the complete history list. -d offset deletes the history entry at the OFFSET position. -a appends history lines.
Super User	
Be a Super User	sudo su
Run command with super user ability	sudo <command>
User Management	
create a user account	sudo useradd <option> <username> option: -m to make a directory for the user
Set password to a user account	sudo passwd <username>
Switch User Account	su <username>
Logout from a user account	exit
Delete user account	sudo userdel <username>
Change user account name	sudo usermod -l <new_name> <old_name>
Current username	whoami
Group Management	
Add group account	groupadd [name]
Group property	grep [group name] /etc/group
Group admin property	grep [group name] /etc/gshadow
Delete group	groupdel [groupname]
Add and Remove members	gpasswd [options] [username] [groupname] -a add single member -M add multiple members -d remove member
make group admin	gpasswd -A [usename] [groupname]
Edit file	
Vim editor	vim <file name>
Nano editor	nano <file name>

Change File Permission	
Change file permission	chmod <permission> <file name> u (Owner) - Permissions used for the owner of the file. g (Group) - Permissions used by members of the group. o (Other) - Permissions used by all other users. r (read) – permit to read the file. w (write) – permit to write the file. x (execute) – permit to execute the file.
Change File Owner	
Change the file owner	chown [option] owner[:group] file(s)
Change Group Owner	
change the group owner	sudo chgrp <group name> <file name>
ACL	
Access control list is a service which is used for providing special permission to specific users and groups to directories and file.	
check acl permission	getfacl <file/directory name>
give acl permission to user	setfacl -m u:<username>:<permission> <file> permission: r w x
removes acl permission of user	setfacl -x u:<username>: <file>
set ACL permission to Group	setfacl -m g:<groupname>:<permission> <file path>
remove acl permission of group	setfacl -x g: <groupname>: <file path>
Remove all ACL permissions	setfacl -b <file path>
Shell Scripting	
Identify interpreter (first line for script)	#!/bin/bash - GNU Bourne-Again Shell #!/bin/sh - The Bourne Shell #!/bin/csh - The C Shell #!/bin/ksh - The Korn Shell
Read input value	read <variable name>
Print	echo / printf
Run file	./<filename>.sh ./file.sh

	bash file.sh sh file.sh
Package manager (may differ by OS)	
Install package	sudo apt-get install <package Name> -y
Update packages	sudo apt-get update
Upgrade packages	sudo apt-get upgrade -y
Manage tools	
Check status of tool	sudo systemctl status <tool name> sudo service <tool name> status
Start Tools	sudo systemctl start <toolname>
Stop tools	sudo systemctl stop <toolname>
Auto enable tools	sudo systemctl enable <toolname>
xargs	
build and execute commands from standard input	xargs [options] [command] echo "New Folder" | xargs mkdir
AWK (prints files data with programming)	
Print	awk [pattern] {action} Pattern: $0 - entire line $1, $2, $3 … - for each column (if exists) NR - number of records (lines) NF - number of fields (columns in the current line) Action: print awk '{print $0}' file.sh
Find (find directory)	
Find the directory	find [option] [path] [expression] option: -name -type -user

	-group -size (MB=M, KB=K, GB=G)
Search the file with less than 10 mb	find /temp -size -10M
Search the file with 10 mb	find /temp -size 10M
Search the file with more than 10 mb	find /temp -size +10M
Find files with guid permission Find files with suid permission Find files with sticky bit permission	find [path] -perm [expression] expressions: 4755 264 1755
Grep (prints specific word in file)	
Find a word in a file(s)	grep [option] [expression] <file name> <file name> … options: -i = intensive (case sensitive) -r = recursively -v = invert string match -l = display the file names that matches the string -L = display the file names that do not contain the string -n =match line with number ^ = display the lines that start with a string
Search and redirect output in a new file	grep [expression] <source file> > <destination file>
Disk space	
To display disk space usage	df [options] [file] -h Human readable -m displays usage in MBs. -k displays usage in KBs. -T shows the file system type in a new column.
Archive	
archives multiple files to TAR format	tar [options] [archive_file] [file or directory to be archived] -c create a new archive -x, –extract, –get extract files from an archive -v Verbose output -f Archive file name -u, –update only append files newer than copy in archive -gzip (gzip compression) -bz2 (compression)

	-J (xz compression) -C (specific destination)
Zip - unzip	zip [options] <zipfile-name> file1 file2…. unzip [option] file_name.zip
File Size	
show file size in human readable format	du -h [file path]
Check running processes	
Linux running processes	top
Interactive program that monitors system resources and server processes in real time	htop
User Manual	
user manual of any commands in terminal	man <command name>
Job Automation	
Two types	at (execute for once) crontab (execute every time)
Set job with at command	at [time] at > [expression] eg. at 8:10 AM at> useradd neelsoni23 ctrl+d
show pending job	atq
remove at job	atrm 2
restrict user accessing at	vim /etc/at.deny neelsoni23 (add username) :wq
Start Crontab service	systemctl start crond
Permanent on crond service	systemctl enable crond
set cron jobs	crontab -e ***** [file path] i. * - Minute field 0 to 59 ii. * - Minute field 0 to 59 iii. * - Day of Month 1-31 iv. * - Month field 1-12 v. * - Day Of Week 0-6
	crontab [options]

	-l jobs of current year -r remove cron jobs -u set for other users
Restrict user from crond service	vim /etc/cron.deny
log file	tail -f /var/log/cron
Show IP	
Show ip address	ifconfig ip addr ip a
SSH	
Remote connect any device with ssh	sudo ssh [options] [username]@[ip address] -i “[key]” for protected key
Services	
services	systemctl [option] [service name] Options: status – display status enable - permanently on disable - permanently off start - start the service stop - to stop restart - to restart e.g systemctl status sshd
Remote File Transfer	
With Secure Copy(scp) File - Local to remote	scp [sourcefile] [username@ip]:[path] scp myfile.txt root@44.33.22.11:/mnt/d/
Folder - Local to remote	scp [option] [sourcefolder] [username@ip]:[path] scp -r /DevOps root@44.33.22.11:/mnt/d/
File - Remote to Local	scp [username@ip]:[path] [sourcefile] scp root@44.33.22.11:/mnt/d/ myfile.txt
Folder - Remote to Local	scp -r [username@ip]:[path] [sourcefolder] scp -r root@44.33.22.11:/mnt/d/ /home/DevOps
With rsync Local to Remote	rsync -rvh [source path] [username@ip]:[destination path] rsync -rvh /home/DevOps root@44.33.22.11:/mnt/d/
Remote to Local	rsync -rvh [username@ip]:[destination path] [source path]

	rsync -rvh root@44.33.22.11:/mnt/d/ home/DevOps