A tour of Open Source Softwares’

INTRODUCING FREE AND OPEN SOURCE SOFTWARE LAB

Open Source Software

- Open source software is a computer software whose source code is available openly in internet
- Programmers can modify it to add new features and capabilities without any cost.
- Get it for free of cost.

Android

Ubuntu

Firefox

Open Office

Proprietary Software

- Created by a closed group of people
- source codes are not publicly available
- only the company which has created can modify it.
- We pay for using it

- Examples are :
  - Windows
  - MacOS
  - Internet Explorer
  - Google earth
  - Microsoft Office etc.

OPEN SOURCE SOFTWARE VS PROPIETIARY SOFTWARE

Difference Between OSS Vs PS

- **Assignment No. 1 :**
- **Q1. Submit the Differences between OSS Vs PS**

Public Vs Private

Licence Redistribution

AnywhereVs Specific Platforms

Bug Fixing

Flexibility

Some Popular OSS

- **Assignement No. 1 :**

- **Q2. List atleast 20 OSS.**
- **Q3. Visit any 5 websites and compare the features.**

Linux

MangoDb

Abiquo

GitHub

Explore GitHub

- Visit GitHub
- Explore the projects being carried out.
- Your main task is to understand any one project, find a bug and fix it.
- You would be required to present this work periodically.
- This would contribute to your assessment.

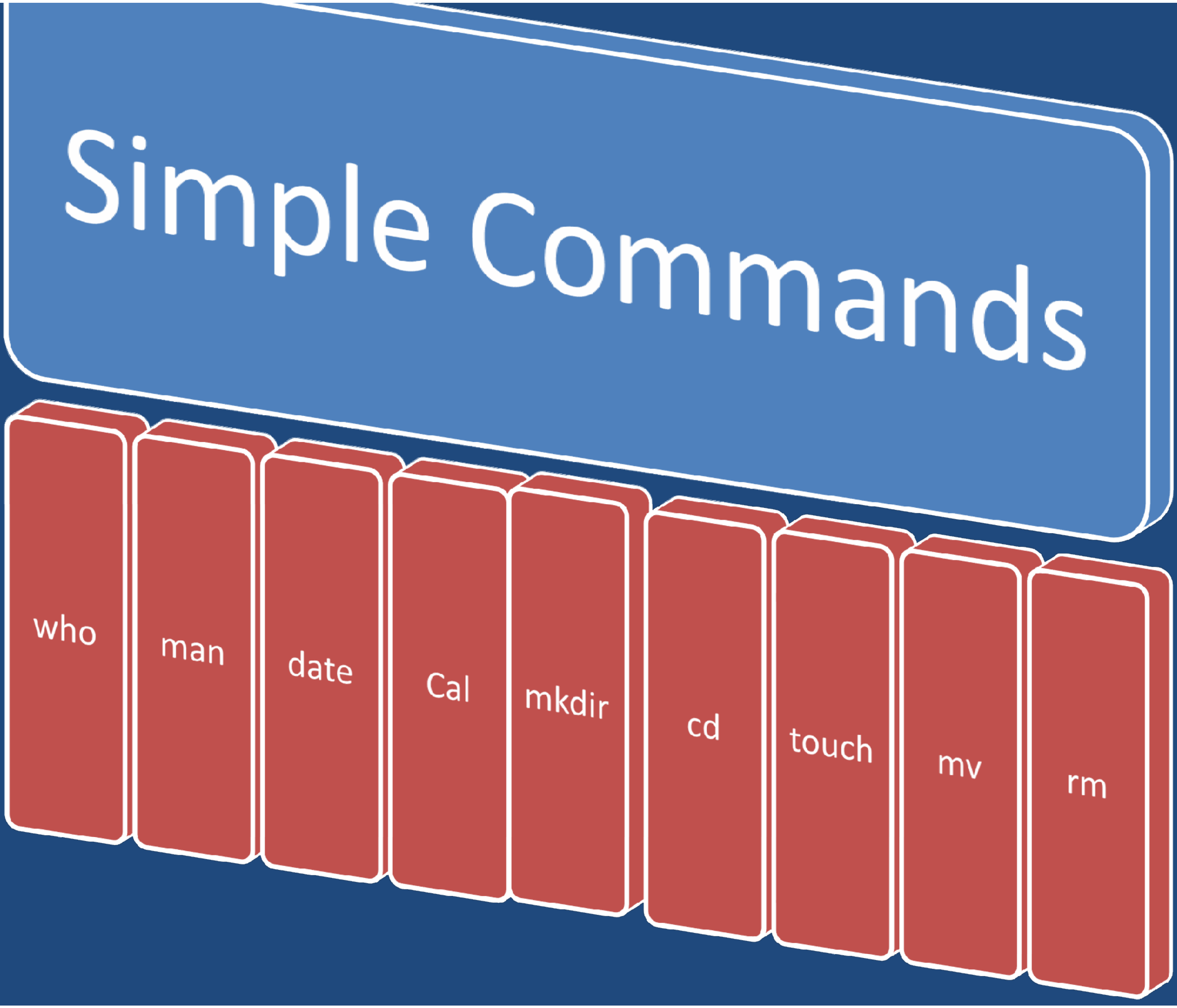
github.com

LINUX : OUR FOCUS

Linux

- The best-known and most-used open source operating system.
- As an operating system, Linux is software that sits underneath all of the other software on a computer, receiving requests from those programs and relaying these requests to the computer's hardware.

https://itsfoss.com/online-linux-terminals/



Simple Commands

- who
- man
- date
- Cal
- mkdir
- cd
- touch
- mv
- rm

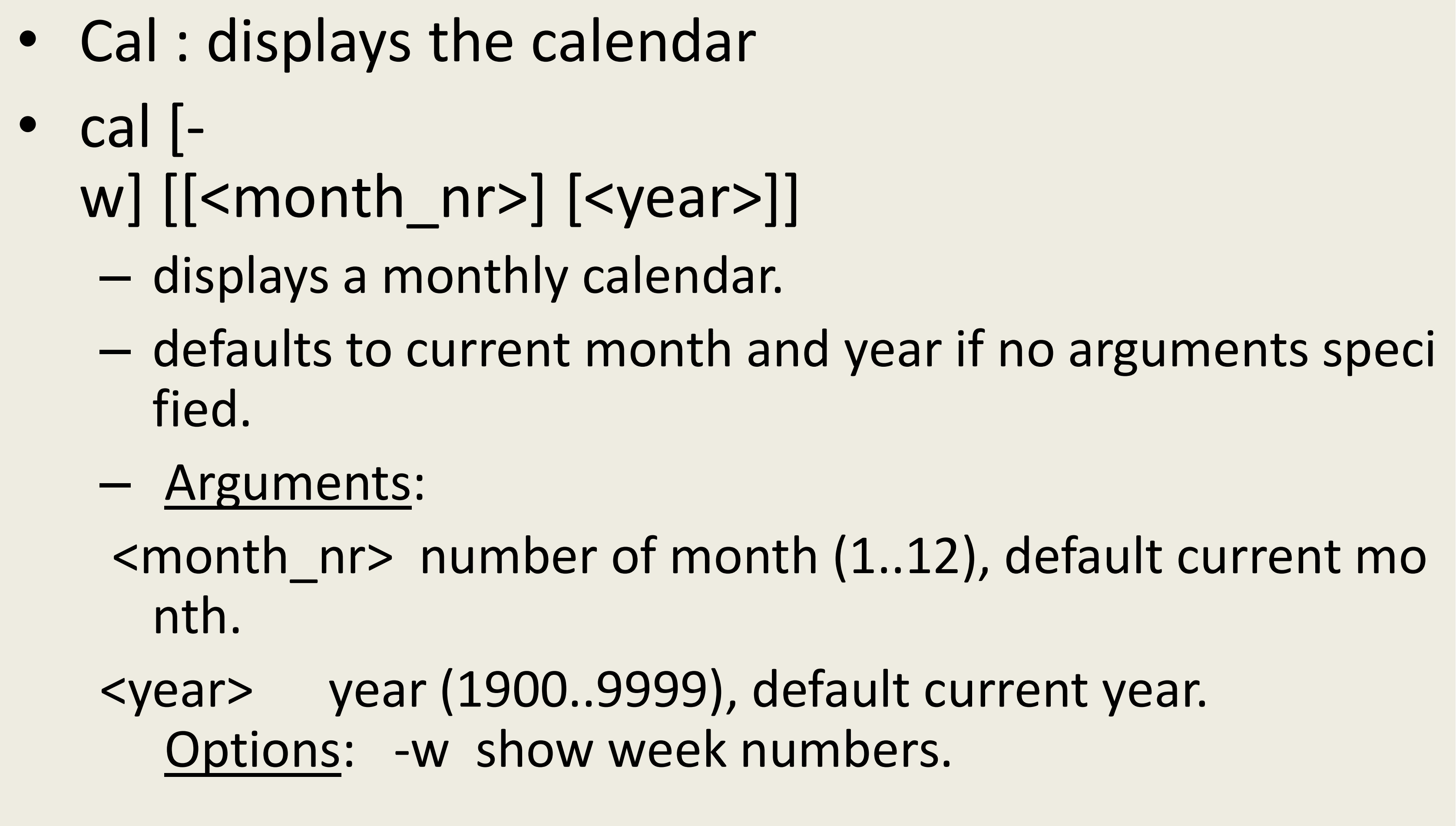
man

- displays a manual page for system
- commands.
- if an entry for the command is found, it will be displayed using the standard pager.
- $ man-p opens a full list of commands
- For example : man cat /displays all the information regarding the cat command and also the various options that can be used with a particular command so as to alter the behavior of the command.

https://www.masswerk.at/jsuix/index.html

https://copy.sh/v86/?profile=linux26

CAL

- 

  - Cal : displays the calendar
  - cal [-w] [[<month_nr>] [<year>]]
    - displays a monthly calendar.
    - defaults to current month and year if no arguments specified.
    - Arguments:

  <month_nr> number of month (1..12), default current month.

  <year>   year (1900..9999), default current year.

  Options: -w show week numbers.

Date : Displays the date and time (Local /UTC)

Arguments	**Syntax : $date +%argument%**
%a	week-day abrv., Sun-Sat
%d	day, 1-31
%D	date as mm/dd/yy
%h	month abrv., Jan-Dec
%H	hours, 00-23
%j	year-day, 001-366
%m	month, 01-12
%M	minutes, 00-59
%n	new line
%r	time in AM/PM
%S	seconds, 00-59
%t	tab (insert space)
%T	time as hh:mm:ss
%w	week-day, 0-6, Sun=0
%y	last two digits of the year 00-99

Try Various Combinations Now

- for example
- date –l // gives local date and time
- Date -u // gives UTC date and time
- date +%D%t%T

Command	Description	Example
Mkdir directory_name	To create a directory	mkdir djg
cd directory path	To change the path of directory To know present working directory	cd djg or Cd/ cd .. (move to parent dir) pwd
touch file name	To create empty files	touch f1 / it will create an empty file named f1. you can create multiple empty files together.
cat >file_name press enter and add content Ctrl+D	To add content to the file:	cat>f1 Welcome to Ctrl+D
Cat file_name	it will display the data of the file f1 to the standard output	cat f1
Cat f1>>f2	For appending the data into a	cat f1>>f2 / append the

Command	Description	example
rm file_name	it will remove the file named f1. you can remove multiple files at the same time.	Rm f1
cp source file destination file	copy the content from one file to another	Cp f1 f2 Contents of file f1 will be copied to file 2
mv source file destination file	To move the content of one file to another file. After moving the content, the source file will not exist.	Mv f1 f2 it will move the content of file f1 to file f2.
Rm directory_name	To remove directory Provided directory is empty.	rmdir djg

Try These Now: Exercises

- How to create more than one directory instead of invoking mkdir multiple(three) times-like? Mkdir -v dir1/dir2/dir3
- What’s the difference between mkdir -v dir1 /dir2/dir3 and mkdir -vp dir1/dir2/dir3? Later creates subdirectories
- How to see sub-directories created mkdir -vp dir1/dir2/dir3? ls –R
- How to move to parent directory? Cd ..
- How to check current working directory? pwd
- How to move to previous working directory? Cd –
- How to list directory contents without using ls command? Dir
- To display a message on the screen? Echo hello
- To redirect echo message to a file : echo hello > hello.txt
- To append data to already existing file : echo “Linux” >>hello.txt

- To view only top two lines of a file.
- To view last two lines of a file
- To check the stats of the files and directories we have create so far . Stat hello.txt or stat dir1 observe the differences
- To display the disk usage of current directory. Du
  - Try du -h : to output in a human readable format
  - Du –max-depth 3
- Create a file hello.txt in dir1. Copy hello.txt into dir2 at the same time, rename it as “bye.txt".
  - Cp hello.txt dir2/bye.txt
- Copy all files ending with ".txt" from dir2 into dir3 which is in dir2.
  - Cp dir2/*.txt dir2/dir3
- To detect accidental or deliberate file corruption while moving from one folder or machine to another.
  - Md5sum hello.txt both at source and destination

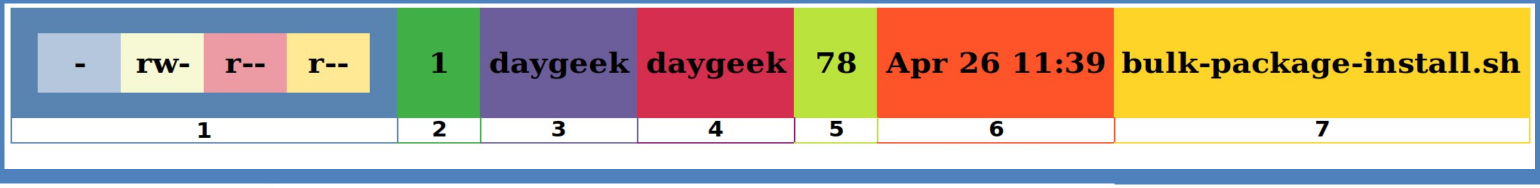
- To remove the directory without deleting its files exclusively : rm -rf directoryname /
- Create a file with some contents with echo.
- echo -e "col1 col2 r1\ncol5 col6 r2\ncol3 col4 r3 " >> new.txt
- Cut a file : cut –f1 –d ‘ ’ new.txt
  - Here f specifies the col. No. (1 in this case) and d is teh delimiter
- paste merges the lines of files : paste hello.txt new.txt
- To compare two files : diff hello.txt new.txt

ls command

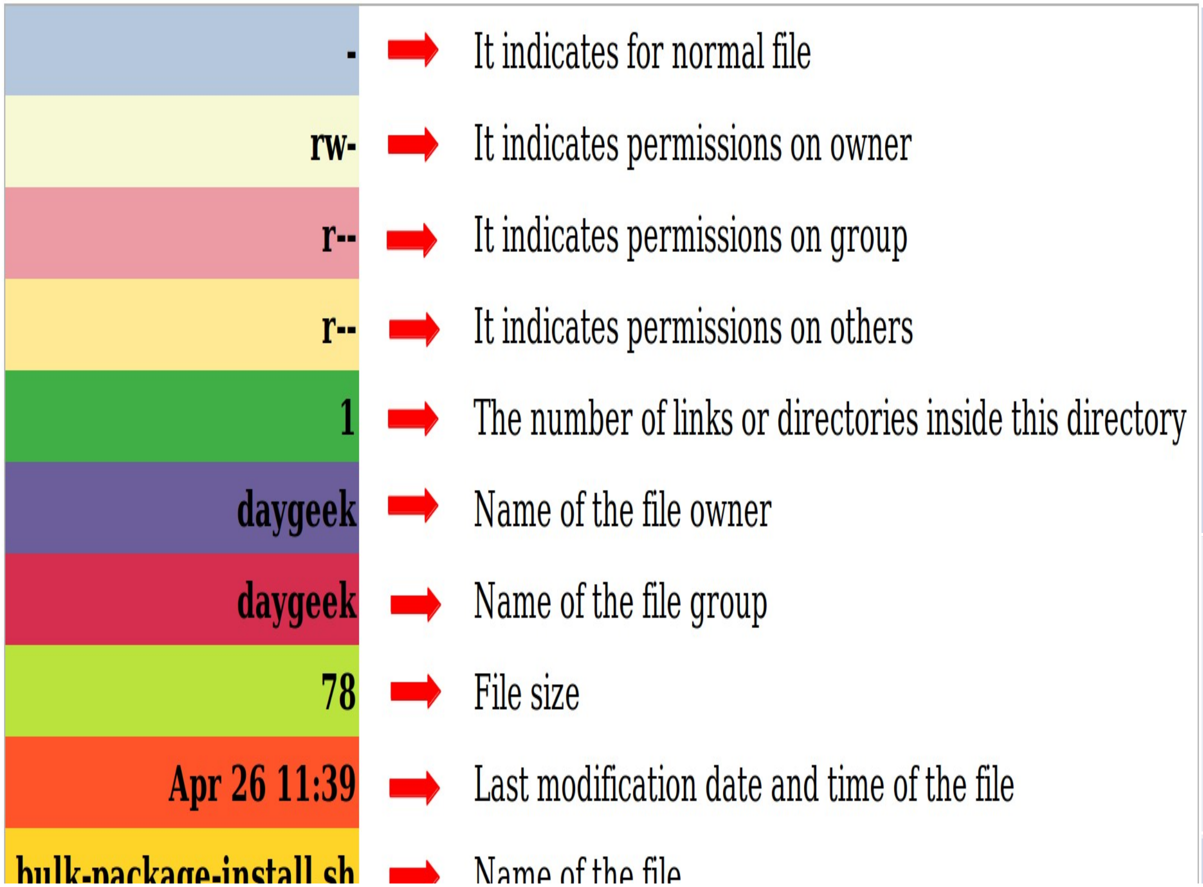
- **stands for list**
- is used to display the directory contents.
- It lists bunch of information about files and directories such as file permissions, number of links, owner name, owner group, file size, time of last modification, and file/directory name.

ls -l

The ls command output comes with seven fields.



- rw- r-- r--	1	daygeek	daygeek	78	Apr 26 11:39	bulk-package-install.sh
1	2	3	4	5	6	7



-	➡ It indicates for normal file
rw-	➡ It indicates permissions on owner
r--	➡ It indicates permissions on group
r--	➡ It indicates permissions on others
1	➡ The number of links or directories inside this directory
daygeek	➡ Name of the file owner
daygeek	➡ Name of the file group
78	➡ File size
Apr 26 11:39	➡ Last modification date and time of the file
bulk-package-install.sh	➡ Name of the file

d : It indicates for

l : It indicates for link

s : It indicates for

Command and options	Description
Ls -a	To view hidden files
Ls -lh	List Files with Human Readable Format (in terms of size)
Ls -F	List Directories with ‘/’ Character at the end
Ls -r	List Files in Reverse Order
Ls –ls (LS is different)	Shows output by file size
Ls -i	Display Inode number of File or Directory
Ls --version	Shows version of ls command

Practice Time

- List Directory Contents Using ls Command Without Any Option
- How to List Directory Contents With Detailed Format?
- How to List Directory Contents In Human Readable Format?
- How to List Hidden Files And Directories From the Given Directory In Linux?
- How to Sort Files and Directories Based on Last Modified Time In Linux?
- How to Sort Files and Directories Based on Last Modified Time in Reverse Order on Linux?
- How to List Only Directory Contents With Vertical Format When You Use ls Command Without Any Option?
- How to List Directory Contents In Reverse Order On Linux?
- How to List Directory Contents Recursively In Linux Using ls Command?
- How to Sort Directory Contents Based on File Size In Linux Using ls Command?
- How to List Directory Contents With INODE Number In Linux Using ls Command?
- How to List Directory Contents With UID and GID In Linux Using ls Command?
- How to Check Directory Permission In Linux Using ls Command?
- How to List Only Hidden Files And Directories From the Given Directory In Linux?

- List Directory Contents Using ls Command Without Any Option : ls
- How to List Directory Contents With Detailed Format? ls -l
- How to List Directory Contents In Human Readable Format? Ls -lh
- How to List Hidden Files And Directories From the Given Directory In Linux? Ls –a or ls -lah
- How to Sort Files and Directories Based on Last Modified Time In Linux? Ls –t or Ls -lth
- How to Sort Files and Directories Based on Last Modified Time in Reverse Order on Linux? Ls -ltrh
- How to List Only Directory Contents With Vertical Format When You Use ls Command Without Any Option? Ls -1
- How to List Directory Contents In Reverse Order On Linux? Ls -lhr
- How to List Directory Contents Recursively In Linux Using ls Command? Ls -R
- How to Sort Directory Contents Based on File Size In Linux Using ls Command? Ls –S or ls -lhS
- How to List Directory Contents With INODE Number In Linux Using ls Command? Ls –i or ls -lhi
- How to List Directory Contents With UID and GID In Linux Using ls Command? Ls -lhn
- How to Check Directory Permission In Linux Using ls Command? ls -ld djg/
- How to List Only Hidden Files And Directories From the Given Directory In Linux? ls –ld .* or ls –ld .?*

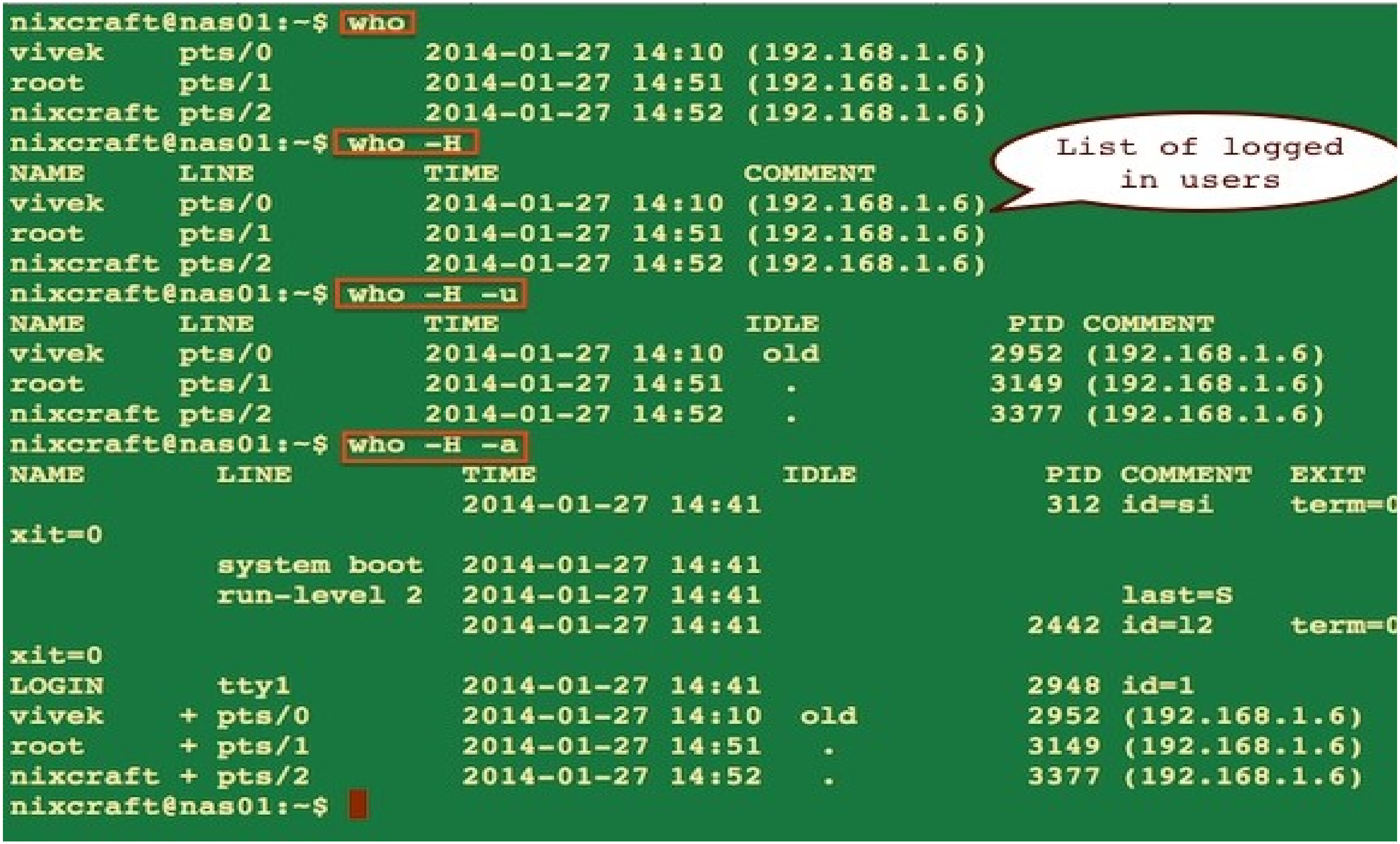
who

- it will list all the users that are currently logged into the system along with
  - their user name,
  - terminal name and
  - date and time of login

Who command options

Option	Description
-b	Time of last system boot
-d	Print dead processes
-H	Print line of column headings
-l	Print system login processes
-m	Only hostname and user associated with stdin
-p	Print active processes spawned by init
-q	All login names and number of users logged on
-r	Print current runlevel
-t	Print last system clock change
-T	Add user’s message status as +, – or ?
-u	List users logged in
-a	Same as -b -d –login -p -r -t -T -u

Who output (With or without options)



```
nixcraft@nas01:~$ who
vivek    pts/0        2014-01-27 14:10 (192.168.1.6)
root     pts/1        2014-01-27 14:51 (192.168.1.6)
nixcraft pts/2        2014-01-27 14:52 (192.168.1.6)
nixcraft@nas01:~$ who -H
NAME     LINE         TIME             COMMENT
vivek    pts/0        2014-01-27 14:10 (192.168.1.6)
root     pts/1        2014-01-27 14:51 (192.168.1.6)
nixcraft pts/2        2014-01-27 14:52 (192.168.1.6)
nixcraft@nas01:~$ who -H -u
NAME     LINE         TIME             IDLE          PID COMMENT
vivek    pts/0        2014-01-27 14:10  old         2952 (192.168.1.6)
root     pts/1        2014-01-27 14:51   .          3149 (192.168.1.6)
nixcraft pts/2        2014-01-27 14:52   .          3377 (192.168.1.6)
nixcraft@nas01:~$ who -H -a
NAME       LINE         TIME             IDLE          PID COMMENT   EXIT
                        2014-01-27 14:41               312 id=si     term=0
xit=0
           system boot  2014-01-27 14:41
           run-level 2  2014-01-27 14:41                   last=S
                        2014-01-27 14:41              2442 id=l2     term=0
xit=0
LOGIN      tty1         2014-01-27 14:41              2948 id=1
vivek    + pts/0        2014-01-27 14:10  old         2952 (192.168.1.6)
root     + pts/1        2014-01-27 14:51   .          3149 (192.168.1.6)
nixcraft + pts/2        2014-01-27 14:52   .          3377 (192.168.1.6)
nixcraft@nas01:~$ 
```

Clearing the Linux Terminal

clear	To Clear Linux terminal, may depend on the terminal emulator you are using
	If you use clear command on Ubuntu with GNOME Terminal, it will clear the screen and you won’t be able to see what else you had on the screen previously.
	In many other terminal emulators, it may just clear the screen for one page. If you scroll with mouse or PageUp and PageDown keys, you can still access the old screen outputs
Ctrl+L	keyboard shortcut in Linux to clear the screen. It works in most terminal emulators.
	Ctrl+L moves the screen one page down giving the illusion of a clean screen but you can still access the command output history by scrolling up.
reset	For clearing the terminal screen. Actually, it performs terminal re-initialization. Takes bit longer than clear command

Create a New **Hidden File** or **Folder**

- Use the mkdir **command** to **create** a new **folder**.
- To **make** that **folder hidden**, add a dot (.) at the beginning of the name

Filters in Linux

- Filters are programs that take plain text(either stored in a file or produced by another program) as standard input, transforms it into a meaningful format, and then returns it as standard output.
- Linux has a number of filters. Some of the most commonly used filters are explained below:

Filter Command	Description	Syntax
cat	Displays the text of the file line by line.	cat filename
Head	Displays the first n lines of the specified text files. If the number of lines is not specified then by default prints first 10 lines.	head -number_of_lines_to_print filename
tail	It works the same way as head, just in reverse order. The only difference in tail is, it returns the lines from bottom to up.	Tail -number_of_lines_to_print filename
sort	Sorts the lines alphabetically by default there are many options available to modify the sorting mechanism	Sort -options filename Will discuss in detail in next section

Filter	Description	Syntax
sed	sed stands for stream editor. It allows us to apply search and replace operation on our data effectively. *sed* is quite an advanced filter	Sed **‘s/search/replace/g’ filename**
nl	nl is used to number the lines of text data.	nl -options filename Options discussed later
uniq	Removes duplicate lines which are adjacent to each other. Limitation: It can only remove continuous duplicate lines To overcome , use it with sort command using pipe operator	Uniq –options filename OR sort filename | uniq options discussed later

Filter	Description	Syntax
wc	gives the number of lines, words and characters, filename in the data. TO DISPLAY THE LONGEST LINE , USE SWITCH -L	wc filename.txt
grep	to search a particular information from a text file Grep –options path Options to be discussed later	grep searchpattern filename.txt OR cat filename.txt | grep searchpattern
tac	Reverse of cat instead of printing from lines 1 through n, it prints lines n through 1.	tac filename.txt
Fmt	text formatter it reformats paragraphs in specified	Cat>file.txt 1. Cat 2.dog 3. lion

Filter	Description	Syntax
pr	converts text files or standard input for printing	Cat filename| pr
tr	tool translates or deletes characters from standard input and writes results to standard output.	tr options set1 set2 For example : $echo "dimple" | tr [:lower:] [:upper:]
setterm	To set background and foreground color	setterm –term linux -back red Setterm –term linux -fore black

SORT command in Linux

- SORT command is used to sort a file. By default, the sort command sorts file assuming the contents are ASCII.
- Lines starting with a number will appear before lines starting with a letter.
- Sorts alphabetically, by default.
- Lines starting with a lowercase letter will appear before lines starting with the same letter in uppercase.

Try These Examples Now

- Create a file student.txt
- Input names of few students in lowercase
- Run sort command : sort filename.txt
- Observe the output

- Create a file mixstudent.txt
- Input names of few students in lowercase and few in uppercase
- Run sort command
- Observe the output

Options with Sort

option	Description	Syntax
-0	to write the **output to a new file**	Sort input.txt>output.txt OR Sort -o output.txt input.txt Cat output.txt
-r	sorts the input file in reverse order i.e. descending order by default.	sort -r input.txt
-n	to sort the file with numeric data present inside.	Create a file with numeric data. sort -n filename.txt
-nr	To sort a file with **numeric data in reverse order**	Create a file with numeric data. sort -nr filename.txt
-k	To sort a table on the **basis of any column number**	create a table with 2 columns Sort –k 2n tablename.txt

Options with Sort

option	Description	Syntax
-c	to check if the **file given is already sorted or not**	sort -c filename.txt It shows line no. if the file is not sorted else if **there is no output then the file is considered to be already sorted .**
-u	To **sort and remove duplicates**	$ sort -u filename.txt

Practice Time

- Create a text file (practice**.txt**) .
- The option ‘**-e**‘ in the below command enables interpretation of backslash and **/n** tells **echo** to write each string to a new line.

-
  ```
  $ echo -e "computer\nmouse\nLAPTOP\ndata\nRedHat\nlaptop\ndebian\nlaptop" > practice.txt
  ```
- Display the contents of practice.txt using cat
- Sort the contents of practice.txt
- Sort the contents of the input file and write it to a file called (**sorted.txt**) and verify the content by using cat command.
- Now sort the contents of text input file in reverse order and redirect output to a file ‘**reversesorted.txt**‘. Also check the content listing of the newly created file.

- Create a new file (**lsl.txt**) at the same location and populate it using the output of ‘**ls -l**‘ for your home directory.
- $ ls –l >lsl.txt
- $ cat lsl.txt
- Sort the contents of file ‘**lsl.txt**‘ on the basis of **2nd column**
- Sort the contents of file ‘**lsl.txt**‘ on the basis of **9th column** (which is the name of the files and folders and is non-numeric).
- It is not always essential to run sort command on a file. We can pipeline it directly on the terminal with actual command.
- Sort and remove duplicates from the text file. Check if the duplicate has been removed or not.
- Create a third file ‘**lsla.txt**‘ at the current location and populate it with the output of ‘**ls -lA**‘ command.
- Sort the contents of two files on standard output in one go.
- Sort, merge and remove duplicates from these two files. sort -u lsl.txt lsla.txt
- Sort the output of ‘**ls -l**‘ command on the basis of field 2,5 (Numeric) and 9 (Non-Numeric).
  - ls -l | sort -t "," -nk2,5 -k9

nl command in Linux

- Command for text formatting and editing.
- To display the lines with line numbers appended before them
- is used for numbering lines, accepting input either from a file or from **STDIN**. It copies each specified file to **STDOUT**, with line numbers appended before the lines.
- Syntax : nl –option filename.txt

Options with nl

option	Description	Syntax
nl	**To display a file with line numbers. Numbers all non-empty lines**	nl file.txt
-b a	**To number all lines (including empty lines also)**	nl –b a filename.txt
-l	**Count multiple, consecutive, non-empty lines as one (** a single logical line) #number specifies how many lines are considered consecutive	1. nl -l number file.txt 2. nl -b a –l NUMBER file.txt
-i	**Override default increment:** The default increment pattern in Linux is 1	nl -i 3 file.txt
-v	**To make the starting line number different:** The default line number is 1.	nl -v 4 file.txt
-s	**Add a string literal after line numbers**	nl –s”...” file.txt

Options with nl

option	Description	Syntax
-w	**Change column for line numbers.** The default column number is 1.	nl -w 2 file.txt nl -w 4 file.txt nl -w 6 fille.txt
-b p	to append line numbers before those lines only that match the given pattern	nl –b pspecifypattern filename.txt
-n	**To print the lines using a different number format** ln -> left-justified, no leading zeros rn -> right-justified, no leading zeros rz -> right-justified with leading zeros	1. nl –n ln file.txt 2. nl –n rn file.txt 3. Nl –n rz file.text

uniq command

- Filters out the repeated lines in a file.
- **uniq** is the tool that helps to detect the adjacent duplicate lines and also deletes the duplicate lines.
- **uniq** filters out the adjacent matching lines from the input file(that is required as an argument) and writes the filtered data to the output file .
- **uniq [OPTION] [INPUT[OUTPUT]]**
- **INPUT** refers to the input file in which repeated lines need to be filtered out and if INPUT isn’t specified then **uniq** reads from the standard input.
- **OUTPUT** refers to the output file in which you can store the filtered output generated by **uniq** command and if OUTPUT isn’t specified then **uniq** writes to the standard output.

Options with uniq

option	Description	Syntax
uniq	Removes duplicate lines which are adjacent to each other. Limitation: It can only remove continuous duplicate lines To overcome , use it with sort command using pipe operator	Uniq filename.txt OR sort filename | uniq
-c	how many times a line was repeated by displaying a number as a prefix with the line	**uniq -c file.txt**
-d	It only prints the repeated lines and not the lines which aren’t repeated.	uniq –d file.txt
-D	It prints all duplicate lines repeatedly.	Uniq –D file.txt

Options with uniq

option	Description	Syntax
-u	It prints only the unique lines.	Uniq -u filename.txt
-f N	this allows the N fields to be skipped while comparing uniqueness of the lines. This option is helpful when the lines are numbered	**uniq -f N file.txt**
-s N	This is similar to -f N option but it skips N characters but not N fields.	uniq –s N file.txt
-w	To limit the comparison to a set number of characters	uniq –w file.txt
-i	It is used to make the comparison case-insensitive	Uniq filename.txt uniq –i filename.txt

Grep -options

option	Description	Syntax
grep	**Search any pattern**	Grep pattern file.txt
-r	To search in entire directory of files	Grep –r ‘pattern’
-i	By default grep is case sensitive (a is not the same as A) but you can ignore case by using the i switch	Grep –i ‘PAttERN’
-n	To di slay line numbers where the matching string is found	Grep –n ‘string’ file.txt
-v	To display lines that don't match the pattern	Grep –v ‘string’ file.txt

Try These Now: Exercises

- How to print the number of line in a file?
- How to sort multiple files?
- Write a command to sort the content input thought the standard input.
- Write a command to Sort the data of the file in reverse, numeric and dictionary order.
- Write a command to merge two sorted file and store the data into third file.
- Write a command to print only the non-redundant data of the file.
- A student record file contains name, roll no, class and the grades corresponding to each student. Now you have to retrieve only the name and roll no of the students.
- Write a command for pattern searching.
- Exercise the use of meta characters in grep.

Changing File Access Permissions

option	Description	Syntax
Stat	**To check the status of all files**	Stat file.txt
chmod	To change the file access permissions 666 : set the file to world writable. This means the owner, group and others can read and write into file	Chmod -v 666 file1.txt Or chmod a+rw file1.txt
U+rw	only owner can read or write into this file	Chmod u+rw file.txt
A-rw	Change the mode of file such that no one can read or write into this file, not even it's owner	chmod a-rw file1.txt

Changing File Attributes

command	Description	Syntax
dirname	removes non-directory suffix from path name	dirname dir2/dir3/dir4/hi.txt
basename	directory and suffix from pathname and gives the last entry	basename dir2/dir3/dir4/hi.txt
chown	to change file owner. you can use chown only as root user. Expected O/P : chown: changing ownership of file1.txt: Operation not permitted.	chown root file1.tx t

Locate file and its type

command	Description	Syntax
file	to figure out a file type, determines the type of a file as ASCII text	file filename.txt
whereis	to find the location of a certain file, command will locate source files and binaries	whereis ls whereis stdio.h
which	Assume you have two versions of php or linux or anyother s/w, which only " searches only pre-defined directories shown by echo $PATH.	which php which linux
find	to search a file on any directory, Searches for files in a directory hierarchy.	find ~ -name “filename.txt"
	To find regular files and invoke the file command on the results	find . -type f -exec file '{}' \;

Locate file and its type

command	Description	Syntax
find and file	To find regular files and display their attributes using the ls command	find . -type f -exec ls -l '{}' \;
	To find files over 20 bytes in size and list them out	find ~ -type f -size +20c -exec ls -hl {} \;
Exercise for you : EXPLAIN THE FOLLOWING COMMAND		
find ~ -type f -size +20c -exec cp dir1 {} \;		

System and user details

command	Description	Syntax
uptime	Gives the current time, how long the system has been running, how many users are currently logged on, and the system load averages for the past 1, 5,and 15 minutes.	uptime
date	To know current date and time	date
Who/w	To display who all are logged into the system, (w gives more details)	who or w
mount	provides list of mounted file systems	mount
df	to display free disk space on mounted devices	df –h
free	displays the total amount of free and used physical and swap memory in the system, as well as the buffers used by the kernel.	free –m

Linux Process-Related Basic commands

- What's a process?
- Can you find name of your system?
  - hostname or more precisely **/bin/hostname**
- How did you find the hostname?

•*/bin/hostname* is a file or process?

- its a file, always remember *In Linux everything is file.*

- Now run file */bin/hostname*
- Its a ELF 32-bit executable file. Ignore rest of the output. So /bin/hostname is not a process. Its just an executable file.
- When we say its executable file. /bin/hostname, contains instructions stored in a binary format. These instruction will tell Kernel to do some task or action.

Linux Process Related Basic Commands

- Do cat /bin/hostname
  - got combination readable and unreadable characters. Somewhere within those output, we are telling Kernel to read a file name '/etc/hosts' and search for entry '127.0.0.1'.(which typically points to host machine name) and display that content.
  - When we type an executable '/bin/hostname' on bash prompt, Its content whatever you saw few lines above is loaded into RAM. Kernel follows those instruction and takes action.

Let’s Redefine Process

- "Process is nothing but a file-content which is residing in RAM“
- Our system has lot of processes.
- To view them we can use command
  - ps
- Its similar to 'ls' except that ls will show files on disk. 'ps' will show files on RAM

- "In computing, a process is an instance of a computer program that is being executed.“
  - Here, an instance of a computer program means its residing in RAM and being executed means Kernel follows its instruction.

Understanding ps Command

PID	TTY	TIME	CMD
14807	**pts/15**	00:00:00	sh
14835	**pts/15**	00:00:00	ps

Field	Description
CMD	indicates what's the executable file name currently residing in-memory. Its has unique id,which is called 'pid'.
PID	Each process has unique number assigned to it. Since Kernel needs to know 'who' is responsible for these instruction.
TTY	Its terminal allocated to the process. If Kernel needs to ask something or print something it will use this terminal for this process.
Time	It refers to how time CPU took to execute this process's instruction. Its denoted in hh:mm:ss format

Parent Process

- Can we list the parent process of a process?
- Yes, we can
- type :bash
- type ps
-
  ```
  PID   TTY  TIME   CMD
  14807 pts/15 00:00:00   sh
  19188 pts/15 00:00:00  bash
  19209 pts/15 00:00:00  ps
  ```
- To confirm parent of 19188
  - ps -o ppid 19188

- PID 14807 is the parent of 19188
- This is primary bash shell
- It instructed Kernel to create another Bash shell having is 19188 finally 19209 is the id of 'ps' the command used to print above output.
- To print the name of prent
  - ps –o ppid,cmd 19188

Exercise for You

Can we find the Parent of our primary Bash prompt (i.e. grandparent)?

What will happen if you keep on repeating the above steps?

The init process

- If you keep on repeating finding parents of parents process, you will end up on '1'. which is the parent of all process. Its called init process.

Do you know you who created this process?

Moral of the Story

So our Bash thinks it assign job to 'Super Programmer' Kernel, but that fact is 'Kernel is assigning job to Bash via proxy leader named init

Resetting a root password in Linux

- **Step-1 :** Reboot the system and interrupt the boot loader countdown timer by pressing any key except enter.
- **Step-2 :** Find the entry that is normally booted, and change it to halt execution during the initial ram disk.
  - 2.1. Use the cursor keys to highlight the entry that is normally booted and press e
  - 2.2. Use the cursor to move to the line that has the kernel and kernel arguments. This line normally starts withlinux16 or linuxefi.
  - 2.3. Move the cursor to the end of line by pressing End, and add rd.break.
  - 2.4. Press Ctrl+x to boot using the modified stanza and you will get a command line interface.

SUDO COMMAND IN LINUX

- sudo (**S**uper **U**ser **DO**) command used as a prefix of some command that only superuser are allowed to run.
- If you prefix **“sudo”** with any command, it will run that command with elevated privileges or in other words allow a user with proper permissions to execute a command as another user, such as the superuser.
- This is the equivalent of “run as administrator” option in Windows.
- These users who can use the **sudo** command need to have an entry in the **sudoers** file located at **“/etc/sudoers”**
- By default, sudo requires that users authenticate themselves with a password which is the user’s password, not the root password itself.

Sudo -Options

option	**Description**	**Syntax**
sudo	The **sudo command** allows you to run programs with the security privileges of another user (by default, as the superuser).	sudo adduser username
-V	option causes sudo to print the version number and exit.	sudo –V
-l	option will print out the commands allowed (and forbidden) the user on the current host.	sudo –l
-h or -help	option causes sudo to print a usage message and exit.	sudo –help

Sudo -Options

option	Description	Syntax
-v	sudo will update (validates) the user’s timestamp, prompting for the user’s password if necessary.	Sudo -v
-k	invalidates the user’s timestamp. allow a user to revoke sudo permissions (Temporary Kill)	sudo –k
-K	option is used to remove the user’s timestamp entirely (SURE KILL)	sudo –K
-b	option tells sudo to run the given command in the background. (BACKGROUND option)	sudo -b
-p	option allows you to override the default	

User Management in Linux

- A user is an entity, in a Linux operating system, that can manipulate files and perform several other operations.
- Each user is assigned an ID that is unique for each user in the operating system.
- After installation of the operating system, the **ID 0 is assigned to the root user** and the IDs 1 to 999 (both inclusive) are assigned to the system users and hence the ids for local user begins from 1000 onwards. In a single directory, we can create 60,000 users.

Commands to manage user

- To ***list out all the users in Linux***,
  - awk -F
- Using id command, you can get the ***ID of any username***. Every user has an id assigned to it and the user is identified with the help of this id. By default, this id is also the group id of the user.
  - id username
- To add a user. *useradd command* adds a new user to the directory. The user is given the ID automatically depending on which category it falls in. The username of the user will be as provided by us in the command.
  - sudo useradd username

Commands to manage users

- passwd - to assign a password to a user. After using this command we have to enter the new password for the user and then the password gets updated to the new password.
  - passwd username
- ***Accessing a user configuration file***.
  - cat /etc/passwd
  - This commands prints the data of the configuration file. This file contains information about the user in the format.
  - username : x : user id : user group id : : /home/username : /bin/bash

Commands to manage users

- To **change the user ID for a user**.
  - usermod -u new_id username
  - This command can change the user ID of a user. The user with the given username will be assigned with the new ID given in the command and the old ID will be removed.
  - *Example:* sudo usermod -u 1982 test
- To **Modify the group ID of a user**.
  - usermod -g new_group_id username
  - This command can change the group ID of a user and hence it can even be used to move a user to an already existing group. It will change the group ID of the user whose username is given and sets the group ID as the given new_group_id.
  - *Example:* sudo usermod -g 1005 test

Commands to Manage users

- **To change the user login name**
  - using *usermod* command, the old login name of the user is changed to the new login name provided.
  - sudo usermod -l new_login_name old_login_name
  - *Example: sudo usermod -l user1 user2*
- **To change the home directory**.
  - sets the new home directory as the directory whose path is provided.
  - usermod -d new_home_directory_path username
  - *Example:* usermod -d new_home_directory test
- **To delete a user name**.
  - Make sure that the user is not part of a group.
  - If the user is part of a group then it will not be deleted directly, hence we will have to first remove him from the group and then we can delete him.
  - userdel -r username*Example:* sudo userdel -r new_geeks

groupadd command in Linux

- Groups in linux refer to the user groups.
- In Linux, there can be many users of a single system, (normal user can take uid from 1000 to 60000, and one root user (uid 0) and 999 system users (uid 1 to 999)).
- In a scenario where there are many users, there might be some privileges that some users have and some don't, and it becomes difficult to manage all the permissions at the individual user level.
- So using groups, we can group together a number of users, and set privileges and permissions for the entire group.
- ***groupadd*** command is used to create a new user group.
- Syntax : groupadd [option] group_name

- **Example:** sudo groupadd gr1
- Every new group created is registered in the file "*/etc/group*".
- To verify that the group has been created, enter the command : sudo tail /etc/group
- The file shows group information in the following format:
  - group_name : password : group-id : list-of-members

groupadd-Options

option	Description	Example
-f, –force	forces the command to silently abort if the group with given already exists.	Groupadd gr1 groupadd –f gr1
-g, –gid GID	to provide a group id (numeric) to the new group, and it should be non-negative and unique unless explicitly created to be non-unique (using -o option).	If this option is not used, the default id is assigned, which is greater than every other group already present.
-K, –key KEY=VALUE	Overrides /etc/login.defs defaults (GID_MIN, GID_MAX and others).	groupadd -K GID_MIN=500 -K GID_MAX=700
-h or -help	to print a usage message and exit.	Groupadd –help

Try These Options

- **-o, –non-unique :** This option permits to add a group with a non-unique GID.
- **-p, –password PASSWORD :** The encrypted password, as returned by crypt(3). The default is to disable the password. This password is visible to the users. You should make sure the password respects the system’s password policy. The group passwords are stored in “/etc/gshadow” file.
- **-r, –system :** Create a system group. The numeric identifiers of new system groups are chosen in the SYS_GID_MIN-SYS_GID_MAX range, defined in login.defs, instead of GID_MIN and GID_MAX.
- **-R, –root CHROOT_DIR :** Apply changes in the CHROOT_DIR directory and use the configuration files from the CHROOT_DIR directory.

Important Points

- To add a new user into the group, the group is mentioned using *-g* option in the command *useradd*.
  - sudo useradd -g nameofthegroup new_user
- To add an existing user to a group, use the *usermod* command
  - usermod -g nameofthegroup existing_user

Deleting Groups

- *groupdel* command is used to delete a existing group. It will delete all entry that refers to the group, modifies the system account files, and it is handled by superuser or root user.
  - Syntax : groupdel [options] GROUP
- **Files:**
  - **/etc/group** : It contains the account information of the Group.
  - **/etc/gshadow** : It contains the secure group account information.
- **Exit values:** This command exits with the following values.
  - **0:** Success
  - **2:** Invalid Command Syntax.
  - **6:** Specified group doesn’t exist.
  - **8:** Can’t remove users primary group.
  - **10:** Can’t update group file.

groupdel-options

- **-f –force**: It used to delete a group even if it is the primary group of a user.
- **-h –help**: It displays the help message and exit.
- **-R –root**: It apply the changes in the CHROOT_DIR directory. Also, it uses the configuration files from the CHROOT_DIR directory.

Try These Now

- Adding new group to the system.
- Viewing groups and the info related to the group.
- Adding user to group.
- Checking group id.
- Adding multiple users to the at once group.
- Deleting group.
- How to create sudo user?
- Setting root password.
- Getting root access.
- Adding new user to the system.
- Viewing users of the system.
- Viewing user id.
- Setting password of the users.
- Switching users account.
- Disabling user account.
- Deleting user account.