Session 7

# LINUX COMMANDS

Faculty of Engineering & Technology

Ganpat University
॥ विद्यया समाजोत्कर्षः ॥

# Basics of LINUX Command

- Linux is a **Unix-Like operating** system. **All the Linux/Unix commands are** run in the **terminal provided by the Linux** system.
- This terminal is just like the command prompt of Windows OS. Linux/Unix commands are ***case-sensitive***.
- The **terminal can be used to accomplish** all Administrative tasks. This includes **package installation**, **file manipulation**, and **user management**. Linux terminal is user-interactive. The **terminal outputs the results of commands** which are specified by the user itself.
- **Execution of typed command** is done **only after you press the Enter key**.

# LINUX Commands

- **Clear:** This command clears the screen.

  **Syntax:** clear

- **"--help"** : **'--help'** 'command shows usage summary for that command.

  **Syntax**: $date --help

- **$whatis** : It gives one line description about the command. It can be quick reference for any command.

  **Syntax:** $whatis date

# LINUX Commands

- **$man:** The man pages are properly documented pages.

  **Syntax:** $man command name

  $man date

- **WHO** : The who command is used to get information about currently logged in user on to system.

  **Syntax**: $who [options] [filename]

  Without options, who command displays the following information,

  - Login name of users
  - Terminal line number
  - Login time of the users into system
  - Remote host name of user

# LINUX Commands

- **DATE:** date is used to display & set the system date and time.

**Syntax:** $date [options]

$date --date = “string” //Displays the given date string in the format of date.

- For displaying past dates:

```
$date  --date  “2 years ago”
$date  --date  “yesterday”
```

- For displaying future dates:

```
$date  --date   “next tue”
$date  --date  “2 day”
```

- To set Date & time:

```
$date  --set   “Tue Nov 13  15:23:34  PDT 2018”
```

# LINUX Commands

- **Who am i:** Displays the username of the current user when this command is invoked.

  **Syntax:** $whoami [option]

  **Options:** $whoami --help //give help message & exit

  $whoami --version //It gives version information and exit.

- **Cal:** Used to see the calender of a specific month or a whole year. By default, it shows current month's calender as output.

  **Syntax:** cal [[month] year]

  - **For example,**

```
cal

cal 08 2000
cal 2018
cal  -3          //shows calendar of previous, current & next month.
```

# LINUX Commands

- **Echo:** Used to display line/text that are passed as an argument.

  **Syntax:** $echo [option] [string]

- **Note: -e** here enables backslash
  - **echo [string]** as echo "my first program"
  - **echo -e "Geeks \bfor \bGeeks"**
  - Using option '**\b**' – backspace with backslash interpretor '**-e**' which removes all the spaces in between
  - echo -e "Tecmint \nis \na \ncommunity \nof \nLinux \nNerds"
    Using option '**\n**' – New line with backspace interpretor '**-e**' treats new line from where it is used.

  - **\t** : For horizontal tab space
  - **\v** : For vertical tab space

# LINUX Commands

- **ls:** Use the “ls” to list out what files are in the directory.

  **Syntax: $ls**

  **$ls –l** // Lists the files in the working directory in long format.

- **mkdir:** Allows user to create directories. This command create multiple directories at once.

  **Syntax:** mkdir [options] [directories]

  mkdir --version

  mkdir --help

# LINUX Commands

- **cd :** Known as change directory command. Change current working directory.

**Synatx:** **$cd [directory_name]** //To move inside a subdirectories

**$cd /** //Change directory to the root directory

**$cd dir-1/dir-2/dir-3** //This command is used to move inside a directory from directory.

**$cd ~ or $cd** //change directory to the home directory.

**$cd..** //This command is used to move to the parent directory of current directory or one level up from current directory.

# LINUX Commands

- **rmdir:** Used to remove empty directories from file system in Linux.

  **Syntax:** rmdir [-ignore –fail – on -non –empty] directories..

- **pwd:** It prints the path of working directory, starting from the root. pwd stands for print working directory.

  **Syntax**: $pwd

- **cat:** Reads data from file & gives their content as output. It helps us to create, view & concatenates files.

  **Syntax: $cat filename** // It gives content of file

  **$cat file1 file2** //It gives contents of multiple files

  **$cat > new file** //it will create named new file.

# LINUX Commands

- **rm:** Used to remove objects such as files, directories, symbolic links and so on.

  **Syntax:** rm [option] filename

  - **$rm –i d.txt : -i** prompts conformation before delete file. You have to press Y for deletion.
  - **$rm –f e.txt: -f** rm prompts for confirmation removal if any file is write protected. The –f option override this minor operation & removes files forcefully.

- **cp :** Used to copy files or group of files or directory.

  **Syntax:** cp [option] source destination

  cp [option] source directory

  cp [option] source-1 source-2 source-3 source-n Directory

- **mv:** Used to move one or more files or directories from one place to another.

  **Syntax:** mv [option] source destination

# LINUX Commands

- **chmod :** Change the access mode of file.

**Synatx:** chmod [references]  [operator] [mode] file filename

| References | Class | Description |
|---|---|---|
| **u** | Owner | File's owner |
| **g** | Group | Users who are member of file's group |
| **o** | Others | Users who are neither the file's owner nor member of the file's group |
| **A** | All | All three of the above |

| Operators | Description |
|---|---|
| **+** | Adds specified modes to specified class |
| **-** | Removes the specified modes from specified class |
| **=** | The modes specified are to be made the exact modes for the specified classes |

# LINUX Commands

| Modes | **Description** |
|---|---|
| **R** | Permission to read file |
| **W** | Permission to write file |
| **X** | Permission to execute file, or, in case of directory , search it. |

```
chmod o=r file1.txt
```

- **ps:** Used to list the currently running processes & their PIDs along with some other information depends on different options.

  **Synatx:** ps [options]

- **grep**: It stands for **Globally search for regular expression & print out**. It is a filter searches a file for a particular pattern of characters, and display all lines that contain that pattern.

  **Syntax**: grep [options] pattern [files]

  For example, **$grep aaa a1.txt**

# LINUX Commands

| Options | **Description** |
|---|---|
| **-i** | Case sensitive |
| **-c** | Displaying the count number of matches |
| **-l** | Displaying the filenames that matches the pattern |
| **-w** | Checking the whole words in file |
| **-o** | Display only the matched pattern |
| **-n** | Show line number while displaying the output |
| **-v** | Inverting the pattern match. You can display lines that are not matched with given string |

# LINUX Commands

- **To search in multiple files**

**$grep   aaa a1.txt a2.txt**

- To search for the word **phoenix** in all files in the current directory, append ***–w*** to the grep command.
- grep -w abc *

- **To Ignore Case in Grep Searches**

grep -i abc *

- **To Search Subdirectories**
- grep -r phoenix *

# LINUX Commands

- **To Show Lines That Exactly Match a Search String**

```
grep -x “phoenix” *
```

- **To List Names of Matching Files**

```
grep -l phoenix *
```

**To Count the Number of Matches**

```
grep -c phoenix *
```