Linux Fundamentals & Networking in Linux (TCSCCS0204T)

~ by Asst. Prof. Shivkumar R. Chandey,
Department of Computer Science,
Thakur College of Science & Commerce (Autonomous College)

Objectives

- This course introduces various tools and techniques commonly used by Linux programmers, system administrators and end users to achieve their day to day work in Linux environment. It is designed for computer learners who have limited or no previous exposure to Linux.

Unit II Syllabus

Unit II	**Graphical Desktop Session Management** Basic Desktop Operations, Network Management, Installing and Updating Software **Text editors** gedit, vi, vim, emacs, Graphics editors, Multimedia applications. Command Line Command line mode options, Shells, Basic Commands General Purpose Utilities, Installing Software, User management, Environment variables, Command aliases. **Filesystem** Linux Documentation man pages, GNU info, help command More documentation sources File Operations Filesystem, Filesystem architecture, File types, File attributes, Working with files, Backup, compression

Text Books and Reference Books

- Text Books
  - Unix Concepts and Applications by Sumitabha Das, 4th Ed, 2017
  - Official Ubuntu Book, 8th Edition, by Matthew Helmke & Elizabeth K. Joseph with Jose Antonio Rey and Philips Ballew, Prentice Hall
- Additional References
  - Linux kernel Home: http://kernel.org
  - Open Source Initiative: https://opensource.org/
  - The Linux Foundation: http://www.linuxfoundation.org/

Expected Learning Outcomes

1. Upon completion of this course, Learners should have a good working knowledge of Linux, from both a graphical and command line perspective, allowing them to easily use any Linux distribution.

2. This course shall help student to learn advanced subjects in computer science practically.

3. Student shall be able to progress as a Developer or Linux System Administrator using the acquired skill set.

Graphical Desktop Session Management

Basic Desktop Operations

- A whole world of apps
  - Ubuntu offers thousands of apps available for download. Most are available for free and can be installed with just a few clicks.
- Office software
  - Create professional documents, spreadsheets and presentations on Ubuntu with LibreOffice, the open source office suite that's compatible with Microsoft Office. That means you can open and edit files like Word documents, Excel spreadsheets and PowerPoint presentations and share them with other users quickly and easily. You can also use Google docs directly from your desktop.

Basic Desktop Operations

- Web browsing
  - Renowned for speed and security, Ubuntu and Firefox make browsing the web a pleasure again. Ubuntu also support Chrome and other browsers that can be installed from the Ubuntu Software centre.

Basic Desktop Operations

- Email
  - Ubuntu comes with Thunderbird, Mozilla's popular email application, so you'll have fast desktop access to your email. No matter which email services you use; Microsoft Exchange, Gmail, Hotmail, POP or IMAP, email just works.

Basic Desktop Operations

- Photos
  - Ubuntu is full of free apps to help you enjoy, manage, edit and share your photos whatever you use to take them. With fantastic support for cameras and phones, you won’t need any extra drivers to get up and running.

Basic Desktop Operations

- Edit and illustrate
  - Edit your photos or create professional illustrations and designs with tools like Gimp and Krita, available in the Ubuntu Software centre.

Basic Desktop Operations

- Organise your photos
  - With Shotwell, you can quickly and easily import, organise, edit and view your pictures. And you can share your favourite snaps on all popular photo sites and social networks.

Basic Desktop Operations

- Videos
  - Watch HD videos from your browser on Ubuntu, or use the default Movie Player and VLC and OpenShot from the Snap Store. Edit your movies with Shotcut or kdenlive and then watch them in Movie Player.

Basic Desktop Operations

- Gaming
  - From Sudoku to first-person shooters, we've got loads of games that'll keep you busy for hours. There are thousands of games available for Ubuntu, including titles from the Unity and Steam platforms. Pick from critically acclaimed titles such as Dota 2, Kerbal Space Program, Counter Strike: Global Offensive and Borderlands: The Pre-Sequel.

Basic Desktop Operations

- An open source operating system
  - Ubuntu code is shared openly throughout the development cycle. We are transparent about our plans for future releases, so as a developer, hardware manufacturer, or OEM, you can work with us to start building Ubuntu applications and systems now.

Network Management

Linux Networking Commands

- Every computer is connected to some other computer through a network whether internally or externally to exchange some information. This network can be small as some computers connected in your home or office, or can be large or complicated as in large University or the entire Internet.
- Maintaining a system's network is a task of System/Network administrator. Their task includes network configuration and troubleshooting.
- Following is a list of Networking and Troubleshooting commands:

Ifconfig	Display and manipulate route and network interfaces.
ip	It is a replacement of ifconfig command.
traceroute	Network troubleshooting utility.
tracepath	Similar to traceroute but doesn't require root privileges.
ping	To check connectivity between two nodes.
netstat	Display connection information.
ss	It is a replacement of netstat.
dig	Query DNS related information.
nslookup	Find DNS related query.
route	Shows and manipulate IP routing table.
host	Performs DNS lookups.
arp	View or add contents of the kernel's ARP table.
iwconfig	Used to configure wireless network interface.
hostname	To identify a network name.
curl or wget	To download a file from internet.
mtr	Combines ping and tracepath into a single command.
whois	Will tell you about the website's whois.
ifplugstatus	Tells whether a cable is plugged in or not.

Linux ifconfig

- The command ifconfig stands for interface configurator. This command enables us to initialize an interface, assign IP address, enable or disable an interface. It display route and network interface.
- You can view IP address, MAC address and MTU (Maximum Transmission Unit) with ifconfig command.
- A newer version of ifconfig is ip command. ifconfig command works for all the versions.
- Syntax:

```
$ ifconfig
```

Assign IP address and Gateway

- You can assign IP address and Gateway to an interface but these settings will be disabled after system reboot.
- Syntax:

```
$ ifconfig etho <address> netmask <address>
```

Enable or Disable specific interface

- To enable specific interface,

```
$ ifup eth0
```

- To disable specific interface,

```
$ ifdown eth0
```

Installing and Updating Software

- Through Ubuntu Software Center you can manage your softwares.
- If you're using Ubuntu or any Ubuntu flavor, then you already have the Ubuntu Software Center.
- In case your system doesn't have Ubuntu Software Center, run the following command to install the app:

```
sudo apt install ubuntu-software
```

Linux Text Editors

- Linux text editors can be used for editing text files, writing codes, updating user instruction files, and more. A Linux system supports multiple text editors. There are two types of text editors in Linux, which are given below:
  - Command-line text editors such as Vi, nano, pico, and more.
  - GUI text editors such as gedit (for Gnome), Kwrite, and more.
- A text editor plays an important role while coding. So, it is important to select the best text editor. A text editor should not only be simple but also functional and should be good to work with.
- A text editor with IDE features is considered as a good text editor.

Top 20 text editors for Linux

1. Vi/VIM editor
2. Nano editor
3. Gedit editor
4. Sublime text editor
5. VSCode
6. GNU emacs
7. Atom editor
8. Brackets editor
9. Pico editor
10. Bluefish
11. Kate/Kwrite
12. Notepad ++
13. Eclipse
14. gVIM editor
15. Jed editor
16. Geany editor
17. Leaf Pad
18. Light Table
19. Medit text editor
20. CodeLite

1.Vi/VIM editor

- Vim editor is one of the most used and powerful command-line based editor of the Linux system.
- By default, it is supported by most Linux distros. It has enhanced functionalities of the old Unix Vi editor.
- It is a user-friendly editor and provides the same environment for all the Linux distros.
- It is also termed as programmer's editor because most programmers prefer Vi editor.
- Vi editor has some special features such as Vi modes and syntax highlighting that makes it powerful than other text editors. Generally, it has two modes:

1.Vi/VIM editor

- Command Mode: The command mode allows us to perform actions on files. By default, it starts in command mode. In this mode, all types of words are considered as commands. We can execute commands in this mode.

- Insert Mode: The insert mode allows to insert text on files. To switch from command mode to insert mode, press the Esc key to exit from active mode and 'i' key.

- To invoke the vi editor, execute the vi command with the file name as follows:

```
vi <file name>
```

2. Nano editor

- Nano is a straight forward editor. It is designed for both beginners and advanced users. It has many customization features.
- Some advanced features of a nano text editor are as following:
  - It has highly customizable key bindings
  - It supports syntax highlighting
  - It has undo and redo options
  - It provides full line display on the standard output
  - It has pager support to read from standard input

2. Nano editor

- To open file with nano editor, execute the command as follows:

```
nano <file name>
```

- In the nano editor, the useful options are given at the bottom, use the CTRL+ option to perform an operation.
- For example, to exit from the editor, use CTRL +X keys. To learn more about nano editor, Visit Linux Nano Editor.

3. Gedit editor

- Gedit editor is the default editor for the GNOME desktop environment.
- When we open a file, it will open with the Gedit editor. It provides straightforward functionalities like any basic text editor.
- It is a lightweight editor with a straight forward user interface.
- It was publicly released in the year 2000 with a GNOME desktop environment. It is developed using the C programming language and supports all font family.

3. Gedit editor

- Some key features of the gedit text editor are as following:
  - It provides syntax highlighting.
  - It supports internationalized text.
  - It supports several programming languages.
- To invoke the gedit editor from the terminal, execute the below command:

```
gedit <file name>
```

Vi Editor with Commands

- What is vi?
  - The vi editor is elaborated as visual editor.
  - It is installed in every Unix system.
  - It is a very powerful application.
  - An improved version of vi editor is vim.
- To install vim editor:
  - $ sudo apt install vim

To save and quit

- You can save and quit vi editor from command mode. Before writing save or quit command you have to press colon (:). Colon allows you to give instructions to vi.
- exit vi table:

Commands	Action
:wq	Save and quit
:w	Save
:q	Quit
:w fname	Save as fname
ZZ	Save and quit
:q!	Quit discarding changes made
:w!	Save (and write to non-writable file)

Tip

- To exit from vi, first ensure that you are in command mode. Now, type :wq and press enter. It will save and quit vi.
- Type :wq to save and exit the file.
- The vi editor commands are case sensitive.

To switch from command to insert mode

Command	Action
i	Start typing before the current character
I	Start typing at the start of current line
a	Start typing after the current character
A	Start typing at the end of current line
o	Start typing on a new line after the current line
O	Start typing on a new line before the current line

To move around a file

Commands	Action
j	To move down
k	To move up
h	To move left
l	To move right

To jump lines

Commands	Action
G	Will direct you at the last line of the file
``	Will direct you to your last position in the file

To delete

Commands	Action
x	Delete the current character
X	Delete the character before the cursor
r	Replace the current character
xp	Switch two characters
dd	Delete the current line
D	Delete the current line from current character to the end of the line
dG	delete from the current line to the end of the file

To repeat and undo

Commands	Action
u	Undo the last command
.	Repeat the last command

Command to cut, copy and paste

Commands	Action
dd	Delete a line
yy	(yank yank) copy a line
p	Paste after the current line
P	Paste before the current line

Move forward or backward

Commands	Action
w	Move one word forward
b	Move one word backward
<n>w	Move specified number of words forward
dw	Delete one word
yw	Copy one word
<n>dw	Delete specified number of words

Emacs

- The Emacs is referred to a family of editors, which means it has many versions or flavors or iterations.
- The most commonly used version of Emacs editor is GNU Emacs and was created by Richard Stallman.
- The main difference between text editors like vi, vim, nano, and the Emacs is that is faster, powerful, and simple in terms of usage because of its simple user interface.
- Unlike the vi editor, the Emacs editor does not use an insert mode, and it is by default in editing mode, i.e., whatever you type will directly be written to the buffer, unless you manually enter command mode by using keyboard shortcuts.
- Installing the Emacs Editor:
  - sudo apt-get install emacs

Best Photo Image Editors for Linux

1. GIMP
2. Krita
3. Pinta
4. DigiKam
5. ShowFOTO
6. RawTherapee, an many more.

- Click on the following link to get more details about the Graphic Editors:
  - https://www.tecmint.com/best-image-photo-editors-for-linux/

Multimedia applications

- Five best Linux multimedia applications:
  1. VLC
  2. Amarok
  3. Rhythmbox
  4. Audacity
  5. OpenShot
- For more details on applications and their descriptions:
  - https://www.linuxtechi.com/10-multimedia-software-for-linux-desktop/

Linux User Management

- User management includes everything from creating a user to deleting a user on your system. User management can be done in three ways on a Linux system.
- Graphical tools are easy and suitable for new users, as it makes sure you'll not run into any trouble.
- Command line tools includes commands like useradd, userdel, passwd, etc. These are mostly used by the server administrators.
- Third and very rare tool is to edit the local configuration files directly using vi.
  - /etc/passwd
- The local user database in Linux is /etc/passwd directory.

root

- The root user is the superuser and have all the powers for creating a user, deleting a user and can even login with the other user's account. The root user always has userid 0.

useradd

- With useradd commands you can add a user.
- Syntax:
  - $ useradd -m -d /home/<userName> -c "<userName>" <userName>
- Example:
  - $ useradd -m -d /home/xyz -c "xyz" xyz
  - **Note:** Look at the above snapshot, we have created a user xyz along with creating a home directory (-m), setting the name of home directory (-d), and a description (-c).
  - The 'xyz' will receive userid and primary group id.

/etc/default/useradd

- File /etc/default/useradd contains some user default options. The command useradd -D can be used to display this file.
- Syntax:
  - useradd -D

userdel

- To delete a user account userdel command is used.
- Syntax:
- userdel -r <userName>
- userdel -r xyz

To list out all the users in Linux

- Use the awk command with -F option. Here, we are accessing a file and printing only first colum with the help of print $1 and awk.
- awk -F':' '{ print $1}' /etc/passwd

Linux Shell Commands

- In Linux, commands are ways or instructions through which you can instruct your system to do some action. Commands are executed in the command line.
- Syntax:

```
command [option] [argument]
```

- There are some commands which don't have any option or don't accept any argument such as 'clear' and 'pwd'.

clear

- The 'clear' command clears out all the previous commands and outputs from terminal display.

pwd

- The 'pwd' command stands for 'print working directory'. It doesn't accept any option or argument and displays the detail of current working directory.

Types of Commands

- External or built-in commands
- Built-in commands are internal commands that are built-in the shell. Built-in commands are called from the shell and executed directly within the shell itself. You can list all built-in commands with the help of 'help' and 'compgen -b' command. Some example of built-in commands are 'pwd', 'help', 'type', 'set', 'unset', etc.
- External commands are other than built-in commands. These commands are programs which have their own binary and located in the filesystem. These are the commands that your system offer and are totally shell independent. Mostly these commands reside in /bin, /sbin, /usr/sbin.

type command

- Linux 'type' command tell us whether a command given to the shell is a built-in or external command.
- Linux 'type' command also tells whether a command is aliased or not.

Syntax:

```
$ type <command>
```

Example:

```
$ type pwd
$ type cd
$ type man
$ type cat
$ type file
```

Which command

- Linux 'which' command locates the path of a command.
- Syntax:

```
which <command1> <command2> <command3>....
```

- Example:

```
which ls pwd rmdir mkdir cp cd file man
```

Linux aliases

- Linux 'alias' command replaces one string from the shell with another string.
- It is a shell built-in command.
- It converts a complicated command into a simpler command or in other words, it creates a shortcut by replacing it with the simpler one.
- if there are more than one word in the string being aliased.

alias

- Making 'alias' in command line creates a temporary 'alias'. Temporary aliases are only available until you exit the shell. To make permanent 'alias' store it in bash startup files.
- alias syntax:
- alias <newName>=<command>   (To create alias for commands)
- alias <newName>=<'command arg1 arg2....'>   (To create alias for more than one argument)
- alias <newName>=<'/home/sssit/path/...'>   (To create alias by a path)
- Note: There will be no space on either side of (=) sign while typing 'alias' command. Quotes are necessary

Creating an alias

- Creating alias for 'ls-l' command as 'll'
- Syntax:

```
$ alias <newName>=<'command'>
```

- Example:

```
$ alias ll='ls -l'
```

Creating alias for a path

- Syntax:

```
$ alias <newName>=<'/home/sssit/path/...'>
```

- Example:

```
$ alias path='cd /home/sssit/Downloads/sample'
```

removing alias(unalias)

- With the help of 'unalias' command you can remove created alias.
- Syntax:

```
$ unalias <createdAlias>
```

- Example:

```
$ unalias c
```

Filesystem

- In Linux system, everything is a file and if it is not a file, it is a process. A file doesn't include only text files, images and compiled programs but also include partitions, hardware device drivers and directories. Linux consider everything as as file.
- Files are always case sensitive.

Types of Files

- Regular files (-): It contain programs, executable files and text files.
- Directory files (d): It is shown in blue color. It contain list of files.
- Special files
  - Block file (b)
  - Character device file (c)
  - Named pipe file (p)
  - Symbolic link file (l)
  - Socket file (s)

Linux File Commands

Command	Description
file	Determines file type.
touch	Used to create a file.
rm	To remove a file.
cp	To copy a file.
mv	To rename or to move a file.
rename	To rename file.

Linux File System

- A Linux file system is a structured collection of files on a disk drive or a partition. A partition is a segment of memory and contains some specific data. In our machine, there can be various partitions of the memory. Generally, every partition contains a file system.
- The general-purpose computer system needs to store data systematically so that we can easily access the files in less time. It stores the data on hard disks (HDD) or some equivalent storage type. There may be below reasons for maintaining the file system:
  - Primarily the computer saves data to the RAM storage; it may lose the data if it gets turned off. However, there is non-volatile RAM (Flash RAM and SSD) that is available to maintain the data after the power interruption.
  - Data storage is preferred on hard drives as compared to standard RAM as RAM costs more than disk space. The hard disks costs are dropping gradually comparatively the RAM.

Linux File System

- The Linux file system contains the following sections:
  - The root directory (/)
  - A specific data storage format (EXT3, EXT4, BTRFS, XFS and so on)
  - A partition or logical volume having a particular file system.

Linux File System Structure

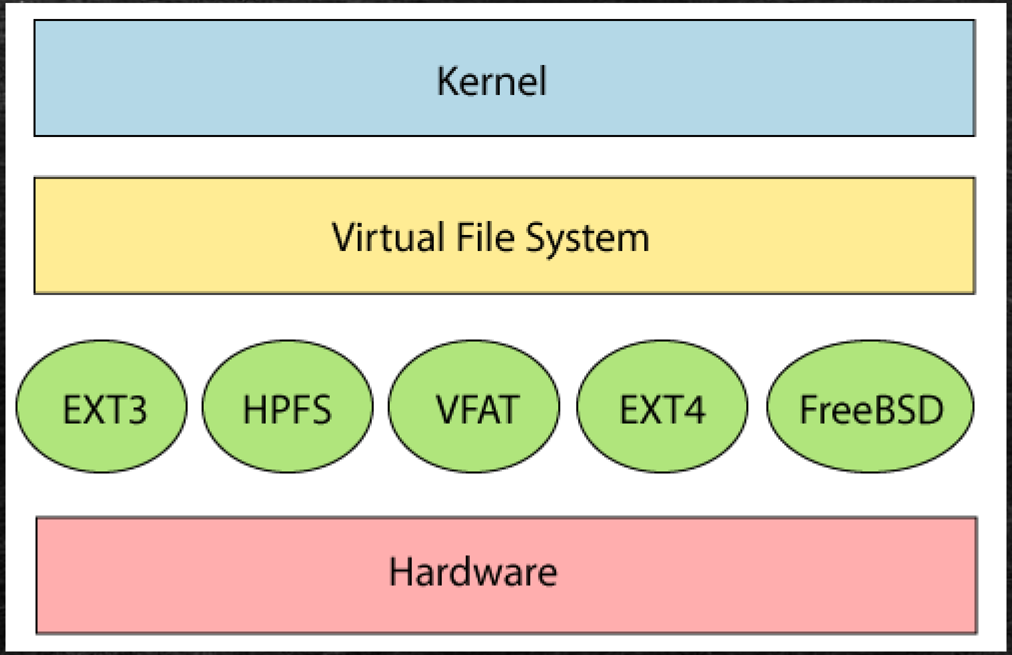
- Linux file system has a hierarchal file structure as it contains a root directory and its subdirectories. All other directories can be accessed from the root directory. A partition usually has only one file system, but it may have more than one file system.
- A file system is designed in a way so that it can manage and provide space for non-volatile storage data. All file systems required a namespace that is a naming and organizational methodology. The namespace defines the naming process, length of the file name, or a subset of characters that can be used for the file name. It also defines the logical structure of files on a memory segment, such as the use of directories for organizing the specific files.
- Once a namespace is described, a Metadata description must be defined for that particular file.

Linux File System Structure

- The data structure needs to support a hierarchical directory structure; this structure is used to describe the available and used disk space for a particular block. It also has the other details about the files such as file size, date & time of creation, update, and last modified.
- Also, it stores advanced information about the section of the disk, such as partitions and volumes.
- The advanced data and the structures that it represents contain the information about the file system stored on the drive; it is distinct and independent of the file system metadata.
- Linux file system contains two-part file system software implementation architecture. Consider the below image:

Linux File System Structure

- Linux File System Structure:



Linux File System Structure

- The Virtual File System (also known as the Virtual Filesystem Switch) is the software layer in the kernel that provides the filesystem interface to userspace programs.
- It also provides an abstraction within the kernel which allows different filesystem implementations to coexist.

Linux File System Structure

- The file system requires an API (Application programming interface) to access the function calls to interact with file system components like files and directories. API facilitates tasks such as creating, deleting, and copying the files. It facilitates an algorithm that defines the arrangement of files on a file system.
- The first two parts of the given file system together called a Linux virtual file system. It provides a single set of commands for the kernel and developers to access the file system. This virtual file system requires the specific system driver to give an interface to the file system.

Linux File System Features

- In Linux, the file system creates a tree structure. All the files are arranged as a tree and its branches. The topmost directory called the root (/) directory. All other directories in Linux can be accessed from the root directory.

- Some key features of Linux file system are as following:

- Specifying paths: Linux does not use the backslash (\) to separate the components; it uses forward slash (/) as an alternative. For example, as in Windows, the data may be stored in C:\ My Documents\ Work, whereas, in Linux, it would be stored in /home/ My Document/ Work.

- Partition, Directories, and Drives: Linux does not use drive letters to organize the drive as Windows does. In Linux, we cannot tell whether we are addressing a partition, a network device, or an "ordinary" directory and a Drive.

Linux File System Features

- Case Sensitivity: Linux file system is case sensitive. It distinguishes between lowercase and uppercase file names. Such as, there is a difference between test.txt and Test.txt in Linux. This rule is also applied for directories and Linux commands.

- File Extensions: In Linux, a file may have the extension '.txt,' but it is not necessary that a file should have a file extension. While working with Shell, it creates some problems for the beginners to differentiate between files and directories. If we use the graphical file manager, it symbolizes the files and folders.

- Hidden files: Linux distinguishes between standard files and hidden files, mostly the configuration files are hidden in Linux OS. Usually, we don't need to access or read the hidden files. The hidden files in Linux are represented by a dot (.) before the file name (e.g., .ignore). To access the files, we need to change the view in the file manager or need to use a specific command in the shell.

Linux Create File

- Linux file system considers everything as a file in Linux; whether it is text file images, partitions, compiled programs, directories, or hardware devices. If it is not a file, then it must be a process. To manage the data, it forms a tree structure.

- Linux files are case sensitive, so test.txt and Test.txt will be considered as two different files. There are multiple ways to create a file in Linux. Some conventional methods are as follows:

Linux Create File

- using cat command
- using touch command
- using redirect '>' symbol
- using echo command
- using printf command
- using a different text editor like vim, nano, vi
- Apart from all of the above methods, we can also create a file from the desktop file manager. Let's understand the above methods in detail:

What is GNU?

- GNU is an operating system that is free software-that is, it respects users' freedom.
- The GNU operating system consists of GNU packages (programs specifically released by the GNU Project) as well as free software released by third parties.
- The development of GNU made it possible to use a computer without software that would trample your freedom.

More about GNU

- GNU is a Unix-like operating system. That means it is a collection of many programs: applications, libraries, developer tools, even games.
- The development of GNU, started in January 1984, is known as the GNU Project. Many of the programs in GNU are released under the auspices of the GNU Project; those we call GNU packages.
- The name “GNU” is a recursive acronym for “GNU's Not Unix.” “GNU” is pronounced g’noo.
- The program in a Unix-like system that allocates machine resources and talks to the hardware is called the “kernel”. GNU is typically used with a kernel called Linux.
- This combination is the GNU/Linux operating system. GNU/Linux is used by millions, though many call it “Linux” by mistake.

Linux Man Command

- The "man" is a short term for manual page. In unix like operating systems such as linux, man is an interface to view the system's reference manual.
- A user can request to display a man page by simply typing man followed by a space and then argument. Here its argument can be a command, utility or function. A manual page associated with each of these arguments is displayed.
- If you will provide a section number in the command, then man will be directed to look into that section number of the manual and that section page will be displayed. And if not, then by default it will display the first page and you have to go through the entire sections in a pre-defined manner.

Syntax of man:

- man [option(s)] keyword(s)

How Output is Displayed in Command Shell

- man display its output through pager. A pager is a program that displays its output one screenfull at a time.
- A colon at the bottom displays end of the on-screen page.
- To go to the next page you can use space bar' or 'f' and to go backward page you can use 'b'.
- To exit from the on-screen page use 'q' and you will be directed to the shell program. And for help press 'h'.

Sections in the man Page

- man page is divided into different sections. Each section is divided based up on a particular topic.
- The man pages have a number written in the parentheses after the command. These number represent the section number. In the above picture, you can see LS(1) at the top, which shows that it is from section 1.
- As I stated earlier, if you will mention a particular section number then your man page will be directed to that section. Or else, you have to go through the whole sections in a pre-defined manner to go to your desired section.

Sections

1. Executable programs and shell commands
2. System calls
3. Library calls
4. Special files
5. File formats and conventions
6. Games
7. Miscellaneous
8. System administration commands
9. Kernel routines

Syntax for a particular section

- Syntax:
- man section_number keyword
- For example

```
man 2 passwd
```

man options

Commands	Function
man -aw	List all available sections of a command.
man -a	To view all man pages of a command.
sman -k (apropos)	Shows a list of results in man page containing a keyword match.
-f, whatis	It displays description from manual page if available.
whereis	Used to determine location of a man page

Different Types of Files in Linux

- A file type helps us in identifying the type of content that is saved in the file.
- Linux supports seven different types of files.
- These file types are the Regular file, Directory file, Link file, Character special file, Block special file, Socket file, and Named pipe file.
- The following table provides a brief description of these file types.

File Types

File type	Description
Ordinary or regular files	Contain data of various content types such as text, script, image, videos, etc.
Directory files	Contain the name and address of other files.
Block or character special files	Represent device files such as hard drives, monitors, etc.
Link files	Point or mirror other files
Socket files	Provide inter-process communication
Named pipe files	Allow processes to send data to other processes or receive data from other processes.

Regular or ordinary files

- Regular or ordinary files store data of various content types such as text, audio, video, images, scripts, and programs. There are hundreds of content types. In Linux, regular files can be created with or without an extension.

- An extension is a group of characters that is used with the file name to give it a special identity or to group it with files of the similar content type. For easy recognition and processing, files of different content types often use well-known file extensions.

- Although the Linux file system does not need file extensions, still you should use them. They help us in identifying the types of content that are stored in files. For example, if a file has a .mp4 extension, you may know that it is a video file.

Regular or ordinary files

- To view a complete list of content types and file extensions that your Linux system supports, you can see the /etc/mime.types file. The MIME (Multipurpose Internet Mail Extensions) provides a standard designation and classification for file content types.
- You can use any standard file reading command to view the content of this file. For example, the following syntax uses the cat command to read this file.

```
$cat /etc/mime.types
```

Directory files

- To organize files in a hierarchy, file systems use directories. Directories are also files, but instead of storing data, they store the location of other files. To store the location of files placed in the directory, the directory uses directory entries. Each directory entry stores the name and location of a single file.
- Linux file system starts with a directory called / or root directory. All files and directory files are created under this directory. Except the root directory, each directory has a parent directory.

Special files

- Linux treats all hardware devices (such as hard drives, printers, monitors, terminal emulators, and CD/DVD drives) as special files. This means that an application program can access and use files and devices in the same way. This feature makes developing programs in Linux easier and flexible.
- Linux places all special files or device files under the /dev directory. There are two types of special files: a character special file and a block special file. A character special file represents a device that transfers data in bytes such as a monitor or a printer. A block special file represents a device that transfers data in blocks such as a hard drive.

Link files

- Link files allow us to use a file with a different filename and from a different location. For this, we use link files. A link file is a pointer to another file. There are two types of links: a hard link and a symbolic or soft link.
- A hard link creates a mirror copy of the original file. A hard link cannot be created to a directory or a file on another filesystem. A soft or symbolic link creates a pointer to the original file. A soft link can be created to a directory or a file on another filesystem.

Socket files

- A socket is a communication endpoint that applications use to exchange data. For example, if an application wants to communicate with another application, it connects with the socket of that application.
- Each application that provides services to other applications or remote clients uses a socket to accept connections. Each socket has an associated IP address and port number that allow it to accept connections from clients.
- For example, if an application of the local system wants to communicate with another application of a remote system, it connects to the socket of that application by using the associated IP address and port number of that socket.

Socket files

- Sockets are very complicated. To make the communication process easier between local applications, Linux uses socket files. Socket files allow applications of the local system to exchange data without going through the complex process of networking and sockets.

- Socket files are the special files that use a file name as their address instead of an IP address and port number. Socket files use the sendmsg() and recvmsg() system calls to enable inter-process communication between local applications.

Named pipe files

- Linux allows us to send the output of any process or command to another process or command as the input. This feature is known as the pipe. Pipes work only when both processes are started by the same user and exist in the same parent process space.

- If processes are executed under different user names and permissions, then standard pipes do not work. In such circumstances, named pipes are used. Named pipes are similar to the standard pipes except that they can be accessed as part of the filesystem.

- Named pipe files are the empty pipe files. The kernel processes named pipe files without writing them to the file system. Named pipe files can exist anywhere in the file system. Named pipe files are also known as the FIFO (First In First Out) files.

File attributes

- File attributes are settings associated with computer files that grant or deny certain rights to how a user or the operating system can access that file. For example, IBM compatible computers running MS-DOS or Microsoft Windows have capabilities of having read, archive, system, and hidden attributes.
- Read-only - Allows a file to be read, but nothing can be written to the file or changed.
- Archive - Tells Windows Backup to back up the file.
- System - System file.
- Hidden - File is not shown when doing a regular dir from DOS.

File attributes

- In operating systems like Linux, there are three main file attributes: read (r), write (w), execute (x).
- Read - Designated as an "r"; allows a file to be read, but nothing can be written to or changed in the file.
- Write - Designated as a "w"; allows a file to be written to and changed.
- Execute - Designated as an "x"; allows a file to be executed by users or the operating system.

Linux gzip

- Gzip (GNU zip) is a compressing tool, which is used to truncate the file size.
- By default original file will be replaced by the compressed file ending with extension (.gz).
- To decompress a file you can use gunzip command and your original file will be back.
- Syntax:

```
gzip <file1> <file2> <file3>. . .
gunzip <file1> <file2> <file3>. . .
```

Linux gzip

- Example:
- gzip file1.txt file2.txt
- gunzip file1.txt file2.txt

gzip options

- Compressing Multi Files Together: If you want to compress more than one file together, you can use 'cat' and gzip command with pipe command.
- Syntax:

```
cat <file1> <file2>. . | gzip > <newFile.gz>
```

- Example:

```
cat file1.txt file2.txt | gzip > final.gz
```

How To Compress A Directory

- The gzip command will not be able to compress a directory because it can only compress a single file. To compress a directory you have to use 'tar' command.
- Hyphen (-) is not mandatory in 'tar' command.
- 'c' is to create,
- 'v' is for verbose, to display output,
- 'f' to mention destination of your output file,
- 'z' for specifying compress with gzip.
- Syntax:

```
tar cf - <directory> | gzip > <directoryName>
```

Popular tools and formats: A tabular comparison

Parameters	gzip	bzip2	p7zip	lzop
Operating systems	Cross-platform	Linux/Unix, Windows	Unix-like	Cross-platform
License	GNU GPL	BSD-like	GNU LGPL	GNU GPL
Compression procedure	Deflate algorithm	Burrows-Wheeler transformation, move-to-front transformation, Huffman coding	LZMA algorithm	LZO algorithm
Data format	.gz	.bz2	.7z	.lzo
Encryption			AES-256	
Compression mode	1–9	1–9	0–9	1, 3, 7–9
Strengths	Very fast	Very good compression rate	Superb compression rate, compresses file directories	Very fast, compresses file directories
Weaknesses	Only compresses single files	Moderate speed, only compresses single files	High system performance demands	Weak compression ra

Google Classroom code for
“Linux Fundamentals & Networking in Linux”

- Join the Google Classroom by Using following Code:

yrtitzh

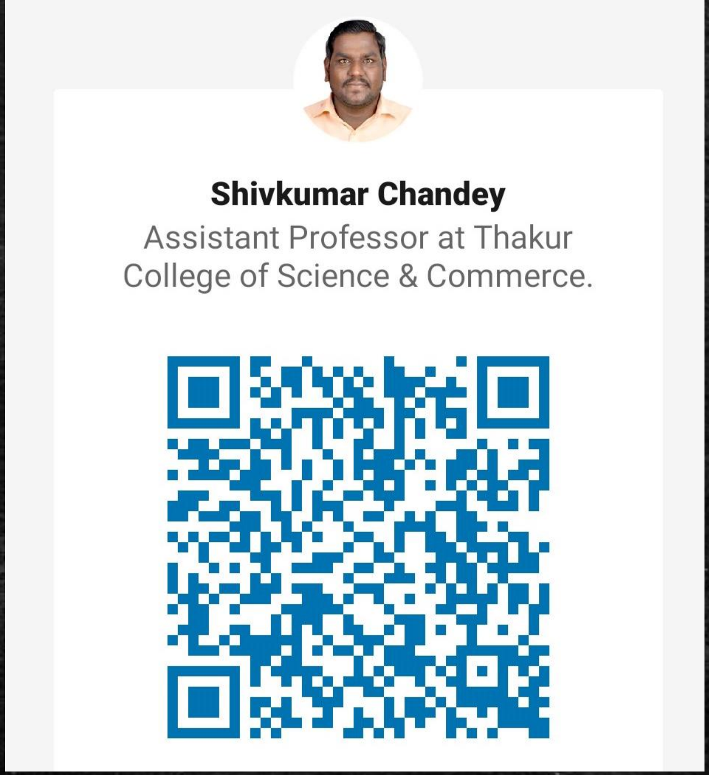
Thank You!!!Any Query?

asktoshivsir@gmail.com

Shivkumar Chandey:

+91 9987389441

Scan QR Code to connect on LinkedIn



Shivkumar Chandey
Assistant Professor at Thakur College of Science & Commerce.