COLLEGE OF ENGINEERING CHENGANNUR

CS232

FREE AND OPEN SOURCE SOFTWARE LAB RECORD

Submitted by

Name AZHAR K

Class S4 D

Roll No. 16

Reg No. CHN17CS037

Certificate

Name : **AZHAR K**

Class : **S4 D**

Roll No : **16**

Exam No :

This is certified to be the bonafide record of practical work done in Free and Open Source Software as per Syllabus of class ..... in the Lab during the academic year 20 /20

.....

Teacher In-charge

.....

Head of Dept.

.....

Examiner's Signature

.....

Principal

Date : .....

Institution Rubber Stamp

CONTENTS

No.	Experiment
1.	Linux Commands
2.	Scripting Task
3.	Networking Task
4.	SSH, Rsync, SCP Task
5.	FTP Task
6.	OS Installation
7.	HTTP & FTP Server Task
8(a).	Package Management
8(b).	PERL
8(c).	LAMP Stack
8(d).	Kernel Compilation
9.	Create Own Webpage

.....

EXPERIMENT NO. :1

LINUX COMMANDS

DATE: 4/02/2019

OBJECTIVE: To familiarize with the basic Linux commands on the terminal.

DESCRIPTION:

Linux is a Unix-Like operating system. All the Linux/Unix commands are run in the terminal provided by the Linux system. The terminal can be used to accomplish all Administrative tasks. The terminal outputs the results of commands which are specified by the user itself.

Following commands were executed on the terminal :

ls - Used to get a list of files and directories

ls -l - Shows file or directory, size, modified date and time, file or folder name and owner of file and its permission.

ls -lt - To order files based on last modified time

ls -ltr - To order files based on last modified time

cat - Used to create single or multiple files, view contain of file, concatenate files and redirect output in terminal or files.

pwd - It prints the path of the working directory, starting from the root.

cd - Used to change the current directory

grep - Processes text line by line and prints any lines which match a specified pattern.

uname - Reports basic information about a computer's software and hardware.

join - To join lines of two files on a common field

RESULT: Familiarized with the various Linux commands and its execution.

EXPERIMENT NO.:2

SCRIPTING TASK

DATE: 11/02/2019

OBJECTIVE: To compute the sgpa of s1 and s2 and calculate cgpa of 2017-2021 batch and display the result of S4D alone with name and register number

DESCRIPTION:

Inorder to compute the sgpa the corresponding file was converted from pdf to text file using pdftotext command.

grep command was used to separate 'CHN17CS' from the file.

Grade and grade points were changed accordingly.

sed command was used for the same.

O,A+,A,B+,B,C,P,F,FE,I
10,9,8.5,8,7,6,5,0,0,0

awk command was used to compute sgpa by finding the average of a column. awk command requires no compiling and can be used to filter strings and text expressions.

The results of S1 and S2 were calculated in two different files and was copied to a third file.The cumulative grade point average was computed with the provided data.

The resulting file displays :

Reg no. Name s1 sgpa s2 sgpa Final cgpa

The following commands were used to carry out this task:

pdftotext/grep/sed/awk/paste/cat

RESULT: Computed the sgpa of S1 and S2 and obtained the cgpa for the respective students.

EXPERIMENT NO.:3

NETWORKING TASK

DATE: 25/02/2019

OBJECTIVE: Set up a network using a network hub.

DESCRIPTION:

A hub, also called a network hub, is a common connection point for devices in a network. Hubs are devices commonly used to connect segments of a LAN. The hub contains multiple ports. When a packet arrives at one port, it is copied to the other ports so that all segments of the LAN can see all packets.

Here a network hub was used to setup a connection and establish an ip. The following commands were used to carry out this task.

ifconfig- ifconfig(interface configuration) command is used to configure the kernel-resident network interfaces.

ifconfig eth0- To view the network settings of an adapter

ifconfig eth0 192.168.0.15 - Assign an ip to an interface eth0

ifconfig eth0 down- Disable an interface

ifconfig eth0 up- Enable an interface

ping- The ping command is used to verify that a device can communicate with another on a network.

RESULT: We connect the router to a hub with a single cable and connect multiple devices to it. Hubs repeat all of the network traffic sent to them. Hence connection is established between two devices.

EXPERIMENT NO.:4

SSH,RSYNC AND SCP

DATE: 25/03/2019

OBJECTIVE: To familiarize with the various data transfer tools and implement the same.

DESCRIPTION:

SSH- Secure Shell is a cryptographic network protocol for operating network services securely over an unsecured network. Secure Shell enables two remotely connected users to perform network communication and other services on top of an unsecured network. It was initially a Unix-based command but is now supported on Windows-based systems as well.

RSYNC- Rsync, which stands for "remote sync", is a remote and local file synchronization tool. It is the most commonly used command for copying and synchronizing files and directories remotely as well as locally in Linux/Unix systems.

SCP- Secure copy protocol (SCP) is a means of securely transferring computer files between a local host and a remote host or between two remote hosts.

The following commands were executed on the terminal:

ssh user_name@host(IP/Domain_name):Used to connect to a remote server or system.

scp source_file_path destination_file_path : used to copy files.

rsync -zvh backup.tar /tmp/backups/ - To copy/sync files and directory locally

rsync -avzh /root/rpmpkgs /tmp/backups/ - To copy/sync a directory on local computer

rsync -avz rpmpkgs/ root@192.168.0.25:/home - To copy/sync files and directory to or from a server.

RESULT: With the help of the following commands files were transferred locally and to the server.

EXPERIMENT NO.:5

FTP TASK

DATE: 11/03/2019

OBJECTIVE: To login to an ftp server with the provided username and password and perform the basic data transfer operations.

DESCRIPTION:

An FTP (File Transfer Protocol) server is often used for data exchanges in many data integration scenarios.It implements a basic FTP client that can send, receive, list, delete files, and create directories. SFTP (Secure File Transfer Protocol) runs over SSH protocol on standard port 22 by default to establish a secure connection.

sftp remote_username@server_ip_or_ostname - To open an sftp connection to a remote system.

sftp> pwd - To check the current working directory

sftp> ls - To list the files and directories.

sftp> get filename.zip - To download a single file from the remote server

sftp> get -r remote_directory - To download a directory from the remote system

sftp> put filename.zip - To upload a file from a local directory to a remote FTP server

sftp> put -r locale_directory - To copy a local directory.

sftp> reput filename.zip - To resume an interrupted upload

RESULT: With the help of the following commands files were uploaded into the remote FTP server .

EXPERIMENT NO.:6

OS INSTALLATION

DATE: 11/03/2019

OBJECTIVE: To install a Linux distro from the provided DVD.

DESCRIPTION:

Fedora 16 was chosen to be installed on the system.Fedora is the open source operating system from the Fedora Project sponsored by Red Hat.

Steps:

1. **Make the PC to boot first from CD/DVD ROM**
2. **Select Start Fedora 16 from the boot screen.**
3. **Start the Installation from Application ——> System Tools ——> Install to Hard Drive.**
4. **Select the installation device type and click Next.**
5. **Select your Time Zone and click Next.**
6. **Enter the password for root account and click Next**
7. **Select your installation type and click Next.**
8. **Confirm the installation**
9. **Once confirmed, fedora 16 will start to install on hard drive.**
10. **Reboot to complete the installation.**
11. **Once rebooted, select the fedora from the GRUB menu**
12. **Click Forward on the Welcome screen.**
13. **Fill the information on the Create User screen.**
14. **Click on Finish.**
15. **Type Password to Login.**

RESULT: Fedora 16 was successfully installed.

EXPERIMENT NO.:7

HTTP&FTP SERVER TASK

DATE: 11/04/2019

OBJECTIVE: To setup http and ftp servers.

DESCRIPTION:

A web server is server software, or hardware dedicated to running said software, that can satisfy World Wide Web client requests

HTTP Server - An HTTP server serves data to clients using the HTTP protocol. It waits for HTTP requests from clients (like your browser) on the Internet and replies with appropriate HTTP responses based on the given HTTP requests.

FTP Server - An FTP Server is a piece of software that is running on a computer and uses the File Transfer Protocol to store and share files.It has has a file transfer protocol (FTP) address and is dedicated to receiving an FTP connection.

An FTP server needs a TCP/IP network for functioning and is dependent on usage of dedicated servers with one or more FTP clients.

The following commands were executed to setup the servers:

sudo apt-get install nginx - To install nginx web server

sudo /usr/sbin/nginx - To start the server

sudo gedit /etc/nginx/sites-enabled-default - Update the configuration files

sudo gedit /var/www/html/index.html - To edit the index.html file

sudo apt install vsftpd - To install vsftpd FTP server

sudo nano /etc/vsftpd.conf - To configure FTP access

ftp -p 192.168.0.189 - To test FTP Access

RESULT: Both HTTP and FTP servers were installed successfully.

EXPERIMENT NO.:8

(a) PACKAGE MANAGEMENT

DATE: 11/04/2019

OBJECTIVE: To install consistent sets of binary packages to the system from the archive.

DESCRIPTION:

Package management is a method of installing and maintaining (which includes updating and probably removing as well) software on the system.The Debian package management system, based on a tool called dpkg with the very popular apt system, is a powerful, popular, and useful method of package management.

STEPS:

sudo apt update - It downloads the package lists from the repositories and "updates" them to get information on the newest versions of packages and their dependencies.

sudo apt upgrade - It will fetch new versions of packages existing on the machine

sudo apt search vlc - It searches for the vlc package to see if its available or not

sudo apt install vlc - To install the vlc package

sudo apt show vlc

sudo apt full-upgrade

sudo apt remove vlc - To uninstall the vlc package

sudo apt purge vlc - It uninstalles the package along with the configuration files.

RESULT: VLC package was installed successfully and later uninstalled with remove and purge.

(b) PERL

DATE: 11/04/2019

OBJECTIVE: To install perl and run the perlscript(perl program).

DESCRIPTION:

Perl stands in for "Practical Extraction and Reporting Language" .Perl is a family of scripting programming languages. It can be compiled just before execution into either code or cross-platform bytecode. Perl includes no. of popular UNIX facilities such as sed, awk, tr etc.Perl is a family of two high-level, general-purpose, interpreted, dynamic programming languages. They provide text processing facilities without the arbitrary data-length limits of many contemporary Unix command line tools,[14] facilitating manipulation of text files.

STEPS:

perl –v – To check if perl is already installed

sudo apt-get install perl - To install perl

touch perlpg.pl - Create a file named perlpg.pl

cat > perlpg.pl - Write the perlscript

perl perlpg.pl - Run the perlscript

RESULT: Following the steps mentioned above perl was successfully installed and the script was run.

(c) LAMP STACK

DATE: 11/04/2019

OBJECTIVE: To install and configure LAMP Stack

DESCRIPTION:

LAMP is an archetypal model of web service stacks, named as an acronym of the names of its original four open-source components: the Linux operating system, the Apache HTTP Server, the MySQL relational database management system, and the PHP programming language.

The Apache web server is among the most popular web servers in the world. It's well-documented and has been in wide use for much of the history of the web, which makes it a great default choice for hosting a website.

sudo apt install apache2 - To install apache using ubuntu package manager

MySQL is a database management system. Basically, it will organize and provide access to databases where your site can store information.

sudo apt install mysql-server - To install mysql

sudo mysql_secure_installation - To run the security script

PHP Hypertext Preprocessor) is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML.

sudo apt install php libapache2-mod-php php-mysql - To install php and related packages

sudo systemctl restart apache2 - To restart the apache web server

sudo gedit /var/www/html/info.php - To create a basic PHP script

RESULT: Apache,MySQL and PHP were installed and a sample PHP script was run.

(d) KERNEL COMPILATION

DATE: 11/04/2019

OBJECTIVE: To build(compile) and install the latest linux kernel from source.

DESCRIPTION:

The various steps involved in building and installing a kernel are as follows:

wget https://cdn.kernel.org/pub/linux/kernel/v4.x/linux-4.20.12.tar.xz - To get the latest Linux kernel source code

unzx -v linux-4.20.12.tar.xzb - Extract tar.xz file

cd linux-4.20.12

cp -v /boot/config-$(uname -r) .config - Configure the Linux kernel features and modules

sudo apt-get install build-essential libncurses-dev bison flex libssl-dev libelf-dev - To install the required compilers and other tools

sudo make modules_install - To install the Linux kernel modules

sudo make install - To install the linux kernel

sudo update-grub - Update the grub config

RESULT: The latest Linux kernel was successfully build and installed.

EXPERIMENT NO.:9

OWN WEBPAGES ON SERVER

DATE: 25/04/2019

OBJECTIVE: To create own webpages on server. Upload contents from our home directory to 192.168.0.30. The pages could be accessed as http://192.168.0.30/cs17d/cs17d16.

DESCRIPTION:

Web pages were created using HTML and the links to different files available at the server was provided along with. A table was created which displays :

- **No.**
- **Experiment description**
- **Code link(typescript)**
- **Date of upload**
- **Github link**
- **Shell script**

HTML tags used for creating the page are:

<table></table> : Used to create a table.
<thead></thead> : Used to make table header.
<tr></tr> : Used to make table row.
<th></th> : Used to make table header data.
<tbody><tbody> : Used to make table body.
<td></td> : Used to make table body data.
** : Used to make link, where href contains the link.**

Uploading index.html to 192.168.0.30 :-

#Commands -:-

(i). sftp cs17d16@192.168.0.30
(ii). put index.html
(iii). bye

The webpage can be accessed by 14.139.189.217/cs17d/cs17d16

Contents of index.html :

```
<html>
	<head>
		<meta http-equiv="content-type" content="text/html; charset=UTF-8">
		<title>cs17d36</title>
	</head>
	<body>
		<h1>S4D CS232 Free and Open Source Software Lab</h1>

		<div>
			<table width="100%" border="1" cellspacing="2" cellpadding="2">
				<thead>
					<tr>
						<th>No</th>
						<th>Experiment Description</th>
						<th>Code Link</th>

						<th>Date of Upload</th>
						<th>Description/Link</th>
						<th>Result/Link</th>
					</tr>
				</thead>
				<tbody>
					<tr>

						<td>1</td>
						<td>Linux Commands</td>
						<td><a href="https://github.com/ceccs17d16/CS232/blob/master/Linux%20Commands/Commands.sh">linux_commands.sh</a></td>
						<td>13/02/2019</td>
						<td><a href="https://github.com/ceccs17d16/CS232/tree/master/Linux%20Commands">files</a></td>
						<td><a href="https://github.com/ceccs17d16/CS232/blob/master/Linux%20Commands/commands1">linux_commands.typescript</a></td>

					</tr>
					<tr>
						<td>2</td>
						<td>CGPA</td>
						<td><a href="https://github.com/ceccs17d16/CS232/blob/master/Scripting%20Task/gpa.sh">shellscript</a></td>
						<td>27/02/19</td>
						<td><a href="https://github.com/ceccs17d16/CS232/tree/master/Scripting%20Task">files</a></td>

						<td><a href="https://github.com/ceccs17d16/CS232/blob/master/Scripting%20Task/gpa">typescript</a></td>
					</tr>
					<tr>
						<td>3</td>
						<td>Networking Task</td>
						<td><a href="https://github.com/ceccs17d16/CS232/blob/master/Networking%20Task/networking.sh">shellscript</a></td>
						<td>28/03/19</td>
						<td><a href="https://github.com/ceccs17d16/CS232/tree/master/Networking%20Task">files</a></td>
					</tr>
					<tr>
						<td>4</td>
						<td>SSH, RSync, SCP</td>
						<td><a href="https://github.com/ceccs17d16/CS232/blob/master/rsync%20ssh%20sftp%20scp/rsync%2Cssh">shellscript</a></td>

						<td>28/03/19</td>
						<td><a href="https://github.com/ceccs17d16/CS232/tree/master/rsync%20ssh%20sftp%20scp">files</a></td>
						<td><a href="https://github.com/ceccs17d16/CS232/blob/master/rsync%20ssh%20sftp%20scp/ssh%2C%20sftp%2C%20scp">typescript</a></td>
					</tr>
					<tr>
						<td>5</td>
						<td>FTP</td>

						<td><a href="">shellscript</a></td>
						<td>28/03/19</td>
						<td><a href="https://github.com/ceccs17d16/CS232/tree/master/rsync%20ssh%20sftp%20scp">files</a></td>
						<td><a href="link">typescript</a></td>
					</tr>
					<tr>
						<td>6</td>

						<td>Linux Installation</td>
						<td>N/A</td>
						<td>10/04/19</td>
						<td><a href="https://github.com/ceccs17d16/CS232/tree/master/Linux%20Installation">files</a></td>
						<td><a href="https://github.com/ceccs17d16/CS232/tree/master/Linux%20Installation">screenshots</a></td>
					</tr>

					<tr>
						<td>7</td>
						<td>HTTP & FTP</td>
						<td>
							<a href="link">N/A</a>
						</td>
						<td>01/04/19</td>

						<td><a href="https://github.com/ceccs17d16/CS232/tree/master/http%20and%20ftp%20server">files</a></td>
						<td>
							<a href="https://github.com/ceccs17d16/CS232/blob/master/http%20and%20ftp%20server/http_server">HTTP_screenshot</a>
							<br>
							<a href="https://github.com/ceccs17d16/CS232/blob/master/http%20and%20ftp%20server/ftp_server">FTP_typescript</a>
						</td>
					</tr>

					<tr>
						<td>8</td>
						<td>Further Tasks</td>
						<td>N/A</td>
						<td>N/A</td>
						<td>N/A</td>

						<td>N/A</td>
					</tr>
					<tr>
						<td>8.a</td>
						<td>Package Management</td>
						<td><a href="https://github.com/ceccs17d16/CS232/blob/master/Further%20Task/Package%20Management/package.sh">shellscript</a></td>
						<td>01/04/19</td>

						<td><a href="https://github.com/ceccs17d16/CS232/tree/master/Further%20Task/Package%20Management">files</a></td>
						<td><a href="https://github.com/ceccs17d16/CS232/blob/master/Further%20Task/Package%20Management/package_management%20Z.txt">typescript</a></td>
					</tr>
					<tr>
						<td>8.b</td>
						<td>Perl</td>
						<td><a href="https://github.com/ceccs17d16/CS232/blob/master/Further%20Task/Perl/add.pl">perlscript</a></td>

						<td>08/04/19</td>
```

```
                        <td><a href="https://github.com/ceccs17d16/CS232/tree/master/Further Task/Perl">files</a></td>
                        <td><a href="https://github.com/ceccs17d16/CS232/tree/master/Further Task/Perl">typescript</a></td>
                    </tr>
                    <tr>
                        <td>8.c</td>
                        <td>LAMP</td>

                        <td>
                            <a href="https://github.com/ceccs17d16/CS232/blob/master/Further%20Task/LAMP/LAMP.sh">shellscript</a>
                            <br>
                            <a href="https://github.com/ceccs17d16/CS232/blob/master/Further%20Task/LAMP/LAMP">typescript</a>
                        </td>
                        <td>22/04/19</td>
                        <td><a href="https://github.com/ceccs17d16/CS232/tree/master/Further%20Task/LAMP">files</a></td>

                        <td>
                            <a href="link">screenshot</a>
                            <br>
                            <a href="https://raw.githubusercontent.com/ceccs17d36/cs232/master/8.further_tasks/LAMP/lamp.typescript">typescript</a>
                        </td>
                    </tr>
                    <tr>
                        <td>8.d</td>

                        <td>Kernel Compilation</td>
                        <td><a href="https://github.com/ceccs17d16/CS232/blob/master/Further%20Task/Kernel%20Compilation/KernelCompilation.sh">shellscript</a></td>
                        <td>23/04/19</td>
                        <td><a href="https://github.com/ceccs17d16/CS232/tree/master/Further%20Task/Kernel%20Compilation">files</a></td>
                        <td><a href="https://github.com/ceccs17d16/CS232/blob/master/Further%20Task/Kernel%20Compilation/KernelCompilation.txt">typescript</a></td>
                    </tr>

                    <tr>
                        <td>9</td>
                        <td>Create Own Server</td>
                        <td>
                            <a href="https://github.com/ceccs17d16/CS232/tree/master/Create%20Own%20Webpage/index.html">index.html</a>
                        </td>
                        <td>22/04/19</td>
                        <td><a href="https://github.com/ceccs17d16/CS232/tree/master/Create%20Own%20Webpage">files</a></td>
                        <td><a href="link"></a></td>
                    </tr>
                </tbody>
            </table>

        </div>
    </body>
</html>
```

RESULT :

The webpage was uploaded into FTP server and was made accessible to public via - 14.139.189.217/cs17d/cs17d16

S4D CS232 Free and Open Source Software Lab

No	Experiment Description	Code Link	Date of Upload	Description/Link	Result/Link
1	Linux Commands	linux_commands.sh	13/02/2019	files	linux_commands.typescript
2	CGPA	shellscript	27/02/19	files	typescript
3	Networking Task	shellscript	28/03/19	files	typescript
4	SSH, RSync, SCP	shellscript	28/03/19	files	typescript
5	FTP	shellscript	28/03/19	files	typescript
6	Linux Installation	N/A	10/04/19	files	screenshots
7	HTTP & FTP	N/A	01/04/19	files	HTTP_screenshot FTP_typescript
8	Further Tasks	N/A	N/A	N/A	N/A
8.a	Package Management	shellscript	01/04/19	files	typescript
8.b	Perl	perlscript	08/04/19	files	typescript
8.c	LAMP	shellscript typescript	22/04/19	files	screenshot typescript
8.d	Kernel Compilation	shellscript	23/04/19	files	typescript
9	Create Own Server	index.html	22/04/19	files	