



CS-506
Unix/Linux

Vision of the Department

To be recognized for keeping innovation, research and excellence abreast of learning in the field of computer science & engineering to cater the global society.

Mission of the Department

- M1:** To provide an exceptional learning environment with academic excellence in the field of computer science and engineering.
- M2:** To facilitate the students for research and innovation in the field of software, hardware and computer applications and nurturing to cater the global society.
- M3:** To establish professional relationships with industrial and research organisations to enable the students to be updated of the recent technological advancements.
- M4:** To groom the learners for being the software professionals catering the needs of modern society with ethics, moral values and full of patriotism.

Program Educational Objectives (PEO's)

- PEO1:** The graduate will have the knowledge and skills of major domains of computer science and engineering in providing solution to real world problems most efficiently.
- PEO2:** The graduate will be able to create and use the modern tools and procedures followed in the software industry in the relevant domain.
- PEO3:** The graduate will be following the ethical practices of the software industry and contributing to the society as a responsible citizen.
- PEO4:** The graduate will have the innovative mindset of learning and implementing the latest developments and research outcomes in the computer hardware and software to keep pace with the fast changing socio economic world.



CS-506
Unix/Linux

COURSE OUTCOMES

- CO1-** Demonstrate installation process for Unix/Linux.
- CO2-** Execute basic commands of Linux OS.
- CO3-** Analyze process states, process scheduling and scheduling priorities.
- CO4-** Implement file creation, file modification and file access permissions
- CO5-** Execute basic Shell Programming assignments

LIST OF EXPERIMENTS

1. To Study basic & User status Unix/Linux Commands.
2. Create a file called wlcc.txt with some lines and display how many lines, words and characters are present in that file.
3. Given two files each of which contains names of students. Create a program to display only those names that are found on both the files.
4. Create a program to find out the inode number of any desired file.
5. Study & use of the Command for changing file permissions.
6. Execute shell commands through VI editor.
7. Installation, Configuration & Customizations of Unix/Linux.
8. Write a shell script that accepts any number of arguments and prints them in the reverse order.
9. Write a shell script to find the smallest of three numbers that are read from the keyboard.
10. Installation of SAMBA, APACHE, TOMCAT.



CS-506
Unix/Linux

Experiment 1

AIM: To Study basic & User status Unix/Linux Commands.

Theory-

1. **who** : The '\$ who' command displays all the users who have logged into the system currently. As shown above, on my system I am the only user currently logged in. The thing tty2 is terminal line the user is using and the next line gives the current date and time

```
$ who
```

Output: harssh tty2 2017-07-18 09:32 (:0)

2. **pwd** : The '\$pwd' command stands for 'print working directory' and as the name says, it displays the directory in which we are currently (directory is same as folder for Windows OS users).

In the output, we are harsshdirectory(folder for Windows OS that are moving to Linux), which is present inside the home directory.

```
$ pwd
```

Output: /home/harssh

3. **mkdir** : The '\$ mkdir' stands for 'make directory' and it creates a new directory. We have used '\$ cd' (which is discussed below) to get into the newly created directory and again on giving '\$ pwd' command, we are displayed with the new 'newfolder' directory.

```
$ mkdirnewfolder
```

```
$ cd newfolder
```

```
$ pwd
```

Output: /home/harssh/newfolder

4. **rmdir** : The '\$ rmdir' command deletes any directory we want to delete and you can remember it by its names 'rmdir' which stands for 'remove directory'.

```
$ rmdirnewfolder
```

5. **cd** : The '\$ cd' command stands for 'change directory' and it changes your current directory to the 'new folder' directory. You can understand this a double-clicking a folder and then you do some stuff in that folder.

```
$ cd new folder (assuming that there is a directory named 'new folder' o
```



CS-506
Unix/Linux

Experiment -1

Aim: To Study basic & User status Unix/Linux Commands

To run basic commands we need open a terminal . you can open it by pressing CTRL + ALT + T or by searching it from start menu or if you want to open from desktop then you can right click then click on "open in terminal ".

- **ls** : If you want to see the list of files on your UNIX or Linux system, use the 'ls' command. It shows the files /directories in your current directory. The ls command will show you the list of files in your current directory

EXPERIMENT NO-02

Aim-Create a file called wlcc.txt with some lines and display how many lines, words and characters are present in that file.

Theory-

Lets consider the below file for our examples.

```
ubuntu@ubuntu:~$ cat inspire.txt.
```

-c Option

Display the number of bytes in the file.

```
ubuntu@ubuntu:~$ wc-c inspire.txt
```

Running the above code gives us the following result:

```
98 inspire.txt
```

-m Option

Display the number of characters in the file.

```
ubuntu@ubuntu:~$ wc-m inspire.txt
```

Running the above code gives us the following result –

```
98 inspire.txt
```



CS-506
Unix/Linux

-w Option

Display the number of words in the file.

```
ubuntu@ubuntu:~$ wc-w inspire.txt
```

Running the above code gives us the following result –

```
15 inspire.txt
```

-l Option

Display the number of lines in the file.

```
ubuntu@ubuntu:~$ wc-l inspire.txt
```

Running the above code gives us the following result:

```
3 inspire.txt
```

-L Option

Display the length of longest line in the file.

```
ubuntu@ubuntu:~$ wc-L inspire.txt
```

Running the above code gives us the following result –

```
38 inspire.txt
```

Only wc

Display the number of lines, words, and characters one after the other.

```
ubuntu@ubuntu:~$ wc inspire.txt
```

Running the above code gives us the following result –

```
3 15 98 inspire.txt
```



CS-506
Unix/Linux

EXPERIMENT NO-03

AIM:-Create a program to find out the inode number of any desired file.
THEORY-

Inode number : Inode number also called index number .An Inode number points to an Inode. An Inode is a data structure that stores the following information about a file :

- Size of file
- Device ID
- User ID of the file
- Group ID of the file
- The file mode information and access privileges for owner, group and others
- File protection flags
- The timestamps for file creation, modification etc
- link counter to determine the number of hard links
- Pointers to the blocks storing file's contents

Check Inodes on Filesystem

You can find a total number of inodes on disk by using '-i' option with df command .

Find Inode number of File: To check inode number of file use following command. The first field in output is an inode number of the file.



CS-506
Unix/Linux

ls -il filename

Inode Changes with Copy, Move and Delete :

Copy file: cp allocates a free inode number and placing a new entry in inode table.

Output :

cp v.txt v_.txt

Move or Rename a file: if destination is same filesystem as the source, Has no impact on inode number, it only changes the time stamps in inode table.

Output :

mv filename path_where_you_want_to_copy

mv v.txt /home/cloudera

Delete a file: Deleting a file in Linux decrements the link count and freeing the inode number to be reused.

Commands to access Inode numbers

- 1) Ls -i Command :the flag -i is used to print the Inode number for each file.
- 2) Df -i Command: df -i command displays the inode information of the file system.
- 3) Stat Command:Stat command is used to display file statistics that also displays inode number of a file.

Output :

ls -i
df -i
stat a.txt



CS-506
Unix/Linux

EXPERIMENT-04

AIM-Study& use of the Command for changing file permissions.

THEORY-

Ownership of Linux files

- **User** : A user is the owner of the file. By default, the person who created a file becomes its owner. Hence, a user is also sometimes called an owner.
- **Group** : A user- group can contain multiple users. All users belonging to a group will have the same access permissions to the file. Suppose you have a project where a number of people require access to a file. Instead of manually assigning permissions to each user, you could add all users to a group, and assign group permission to file such that only this group members and no one else can read or modify the files.



CS-506
Unix/Linux

- **Other** : Any other user who has access to a file. This person has neither created the file, nor he belongs to a usergroup who could own the file. Practically, it means everybody else. Hence, when you set the permission for others, it is also referred as set permissions for the world.

Permissions

- **Read:** This permission give you the authority to open and read a file. Read permission on a directory gives you the ability to lists its content.
- **Write:** The write permission gives you the authority to modify the contents of a file. The write permission on a directory gives you the authority to add, remove and rename files stored in the directory. Consider a scenario where you have to write permission on file but do not have write permission on the directory where the file is stored. You will be able to modify the file contents. But you will not be able to rename, move or remove the file from the directory.
- **Execute:** In Windows, an executable program usually has an extension ".exe" and which you can easily run. In Unix/Linux, you cannot run a program unless the execute permission is set. If the execute permission is not set, you might still be able to see/modify the program code(provided read & write permissions are set), but not run it.

Understanding the security permissions

First, you must think of those nine characters as three sets of three characters (see the box at the bottom). Each of the three “rwx” characters refers to a different operation you can perform on the file.

---	---	---
rwx	rwx	rwx
user	group	other

The ‘r’ means you can “read” the file’s contents.
The ‘w’ means you can “write”, or modify, the file’s contents.
The ‘x’ means you can “execute” the file. This permission is given only if the file is a program.
If any of the “rwx” characters is replaced by a ‘-’, then that permission has been revoked.

Symbolic Mode



CS-506
Unix/Linux

In the Absolute mode, you change permissions for all 3 owners. In the symbolic mode, you can modify permissions of a specific owner. It makes use of mathematical symbols to modify the file permissions.

Operator	Description
+	Adds a permission to a file or directory
-	Removes the permission
=	Sets the permission and overrides the permissions set earlier.

The various owners are represented as -

User Denotations	
u	user/owner
g	group
o	other
a	all

Example

As we know that "ls" is used to list information about files and directories within the file system and "-l" (lowercase L) option causes ls to print files in a long listing format. it shows the following :-

- The file type
- The file permissions
- Number of hard links to the file
- File owner
- File group
- File size



CS-506
Unix/Linux

- Date and Time
- File name

EXPERIMENT NO-05

AIM-Installation, Configuration&Customizations of Unix/Linux

THEORY-

Installing Linux using Virtual Machine



CS-506
Unix/Linux

This is a popular method to install a Linux operating system. The virtual installation offers you the freedom of running Linux on an existing OS already installed on your computer. This means if you have Windows running, then you can just run Linux with a click of a button. Virtual machine software like Oracle VM can install Ubuntu in easy steps. Let us look at them.

Steps to follow

Step 1 : To download virtual box and then click on window host....

Step 2: Click On next

Step 3 : Select you're the directory to install VirtualBox and click on next

Step 4 : Select Desktop icon and click on next, now click on yes

Step 5 : Click On install.

Step 6 : Now installation of the virtual box will start. Once complete, click on Finish Button to start Virtual Box



CS-506
Unix/Linux

AIM- Write a shell script that accepts any number of arguments and prints them in the reverse order.

THEORY-

- \$# Stores the number of command-line arguments that were passed to the shell program.
- \$* Stores all the arguments that were entered on the #command line (\$1 \$2 ...).
- @\$ Stores all the arguments that were entered on the #command line, individually quoted ("\$1"\$2" ...)

Program :

```
echo "input string is :$*"
for (( i=$#;i>0;i-- ));do
    echo "${!i}"
done
```

Program 2

```
echo "input string is :$*"
for x in "$@"
do
y=$x" "$y
done
echo "reversed string is: $y"
```



EXPERIMENT NO-07

AIM- Write a shell script to find the smallest of three numbers that are read from the keyboard.

THEORY-

Program 1

```
echo "enter a: "  
read a  
echo "enter b : "  
read b  
echo "enter c : "  
read c  
s=$a  
if [ $b -lt $s ]  
then  
s=$b  
fi  
if [ $c -lt $s ]  
then  
s=$c  
fi  
echo Smallest of $a $b $c is $s
```

Program 2

```
echo "enter a: "  
read a  
echo "enter b: "  
read b  
echo "enter c: "  
read c  
if [ $a -le $b -a $a -le $c ]  
then  
echo "a is Smallest"  
elif [ $b -le $c -a $b -le $a ]  
then  
echo "b is Smallest"  
else
```



CS-506
Unix/Linux

```
echo "c is Smallest"  
fi  
  
echo "enter a: "  
read a  
echo "enter b: "  
read b  
echo "enter c: "  
read c  
if [ $a -le $b -a $a -le $c ]  
then  
echo "a is Smallest"  
elif [ $b -le $c -a $b -le $a ]  
then  
echo "b is Smallest"  
else  
echo "c is Smallest"  
fi
```



EXPERIMENT NO-08

Write a shell script that accepts any number of arguments and prints them in the reverse order.

```
echo enter n
read n
num=0
while [ $n -gt 0 ]
do
num=$(expr $num \* 10)
k=$(expr $n % 10)
num=$(expr $num + $k)
n=$(expr $n / 10)
done
echo number is $num
```

OUTPUT:

```
$ enter n
$ 456
$ number is 654
```



EXPERIMENT NO-09

Write a shell script to find the smallest of three numbers that are read from the keyboard.

Algorithm:

1. Get three numbers. Say num1, num2, num2
2. If (num1 > num2) and (num1 > num3)
 echo value of num1
3. elif(num2 > num1) and (num2 > num3)
 echo value of num2
4. Otherwise,
 echo value of num3

#shell script to find the greatest of three numbers

```
echo "Enter Num1"  
read num1  
echo "Enter Num2"  
read num2  
echo "Enter Num3"  
read num3  
  
if [ $num1 -gt $num2 ] && [ $num1 -gt $num3 ]  
then  
    echo $num1  
elif [ $num2 -gt $num1 ] && [ $num2 -gt $num3 ]  
then  
    echo $num2  
else  
    echo $num3  
fi
```



CS-506
Unix/Linux

Output

Enter Num1

1

Enter Num2

34

Enter Num3

2

34

Enter Num1

500

Enter Num2

5

Enter Num3

6

500



CS-506
Unix/Linux

EXPERIMENT NO-10

Installation of SAMBA, APACHE, TOMCAT.

Many organizations need to provide network storage and print services for a range of desktop operating systems, Linux uses the samba server to provide services that Microsoft windows clients can use. A Samba server offers the following services:

- 1) Share one or more directory trees
- 2) Share one or more Distributed file-system trees
- 3) Share printers installed on the server among Windows clients on the network
- 4) Assist clients with network browsing
- 5) Authenticate clients logging onto a Windows domain
- 6) Provide or assist with Windows Internet Name Service name-server resolution

Requirements of Samba Server



CS-506
Unix/Linux

- 1) The Samba RPM packages must be installed on our system. Note that there have been no modifications to the Samba RPMs to support high availability.
- 2) The Samba daemons will be started and stopped by the cluster infrastructure on per-service basis. Consequently, the Samba configuration information should not be specified in the conventional `/etc/samba/samba.conf` file. The cluster config tool writes a `smb.conf.surname` file to the `/etc/samba` directory for each samba share
- 3) The automated system startup of the Samba daemons are `smb` and `nmb`. It should be disabled in `init.d` run levels
- 4) Since the cluster infrastructure stops the cluster-related Samba daemons appropriately, do not manually run the conventional Samba stop script (`service smb stop`) as this will terminate all cluster-related samba daemons
- 5) File system mounts for clustered Samba services should not be included in `/etc/fstab`. Rather, for clustered services, the parameters describing mounts are entered via the Cluster Configuration Tool.

Samba Configuration

- 1) Use yum to install Samba package

```
# yum -y install samba
```

- 2) Create the new directory `/smbdemo`

```
# mkdir /smbdemo
```

- 3) change the permission of new directory to `770`

```
# chmod 770 /smbdemo
```

- 4) Change directory



CS-506
Unix/Linux

```
#cd /smbdemo
```

5) Add three empty files to the new directory:

```
# touch file1 file2 file3
```

6) Add Samba user

```
# smbpasswd -a (username)
```

7) Follow the below steps to create a smbusers group, change ownership of the /smbdemo directory, and add a user to the group.

```
[root@localhost smbdemo]# groupadd smbusers
```

```
[root@localhost smbdemo]# chown :smbusers /smbdemo/
```

```
[root@localhost smbdemo]# usermod -G smbusers don
```

8) Configuring samba

Samba configuration file done in the file /etc/samba/smb.conf:

Global settings: This specifies where you configure the server. We can find things like authentication method, ports, workgroup names, and server names.

Share: It specifies where you configure each of the shares for the users.

9) In the global section rename the workgroup to your workgroup name

10) Next add the section for/smbdemo, you can add it very bottom of the configuration file as follows.

```
[smbdemo]
```

```
comment = Linux
```



CS-506
Unix/Linux

path = /smbdemo

browsable = yes

guest ok = yes

Read only = no

create mask = 0755

11) Write and save all changes in the file. And you can test the file using testparm command. When executing the command, the server re-read the file contents. After that you should restart the service smb and nmb. When correctly configured, you can be able to connect from Windows running System and see both general share and the users home directory contents.

Experiment -1

Aim: To Study basic & User status Unix/Linux

Commands

To run basic commands we need open a terminal .



you can open it by pressing CTRL + ALT + T or by searching it from start menu or if you want to open from desktop then you can right click then click on 'ls' command. It shows the files /directories in your current directory. The ls command will show you the list of files in your curr