LAB MANUAL

SYSTEM ADMINISTRATION LAB

SEMESTER 6 : COMPUTER ENGINEERING

EXPERIMENT 1

AIM: To study about linux operating system

DESCRIPTION: Linux is a Unix-like and POSIX-compliant computer operating system assembled under the model of free and open source software development and distribution. The defining component of Linux is the Linux kernel,[an operating system kernel first released on 5 October 1991 by Linus Torvalds. Because it considers Linux to be a variant of the GNUoperating system, initiated in 1983 by Richard Stallman, the Free Software Foundation prefers the name GNU/Linux when referring to the operating system as a whole; see GNU/Linux naming controversy for more details.

Linux was originally developed as a free operating system for Intel x86-based personal computers. It has since been ported to more computer hardware platforms than any other operating system. It is a leading operating system on servers and other big iron systems such as mainframe computers and supercomputers: as of June 2013, more than 95% of the world's 500 fastest supercomputers run some variant of Linux, including all the 44 fastest. Linux also runs on embedded systems (devices where the operating system is typically built into the firmware and highly tailored to the system) such as mobile phones,[tablet computers, network routers, building automation controls, televisions and video game consoles; the Android system in wide use on mobile devices is built on the Linux kernel.

The development of Linux is one of the most prominent examples of free and open source software collaboration: the underlying source code may be used, modified, and distributed — commercially or non-commercially — by anyone under licenses such as the GNU General Public License. Typically, Linux is packaged in a format known as a Linux distribution for desktop and server use. Some popular mainstream Linux distributions include Debian (and its derivatives such as Ubuntu and Linux Mint), Fedora (and its derivatives such as the commercial Red Hat Enterprise Linux and its open. Linux distributions include the Linux kernel, supporting utilities and libraries and usually a large amount of application software to fulfill the distribution's intended use.Linux is freely redistributable, anyone may create a distribution for any intended use.

RESULT: Familiarized with linux operating system

<u>EXPERIMENT 2</u>

<u>AIM:</u> To study the role of a system administrator

<u>DESCRIPTION:</u> A system administrator, or sysadmin, is a person who is responsible for the upkeep, configuration, and reliable operation of computer systems; especially multi-user computers, such as servers.

The system administrator seeks to ensure that the uptime, performance, resources, and security of the computers he or she manages meet the needs of the users, without exceeding the budget.

To meet these needs, a system administrator may acquire, install, or upgrade computer components and software; automate routine tasks; write computer programs; troubleshoot; train and/or supervise staff; and provide technical support.

Many organizations staff other jobs related to system administration. In a larger company, these may all be separate positions within a computer support or Information Services (IS) department. In a smaller group they may be shared by a few sysadmins, or even a single person.

- A database administrator (DBA) maintains a database system, and is responsible for the integrity of the data and the efficiency and performance of the system.
- A network administrator maintains network infrastructure such as switches and routers, and diagnoses problems with these or with the behavior of network-attached computers.
- A security administrator is a specialist in computer and network security, including the administration of security devices such as firewalls, as well as consulting on general security measures.
- A web administrator maintains web server services (such as Apache or IIS) that allow for internal or external access to web sites. Tasks include managing multiple sites, administering security, and configuring necessary components and software. Responsibilities may also include software change management.
- A computer operator performs routine maintenance and upkeep, such as changing backup tapes or replacing failed drives in a RAID. Such tasks usually require physical presence in the room with the computer; and while less skilled than sysadmin tasks require a similar level of trust, since the operator has access to possibly sensitive data.
- A postmaster administers a mail server.
- A Storage (SAN) Administrator. Create, Provision, Add or Remove Storage to/from Computer systems. Storage can be attached local to the system or from a Storage Area Network (SAN) orNetwork Attached Storage (NAS). Create File Systems from newly added storage.

In some organizations, a person may begin as a member of technical support staff or a computer operator, then gain experience on the job to be promoted to a sysadmin position.

Duties

A system administrator's responsibilities might include:

- Analyzing system logs and identifying potential issues with computer systems.
- Introducing and integrating new technologies into existing data center environments.
- Performing routine audits of systems and software.
- Performing backups.
- Applying operating system updates, patches, and configuration changes.
- Installing and configuring new hardware and software.
- Adding, removing, or updating user account information, resetting passwords, etc.
- Answering technical queries and assisting users.
- Responsibility for security.
- Responsibility for documenting the configuration of the system.
- Troubleshooting any reported problems.
- System performance tuning.
- Ensuring that the network infrastructure is up and running.
- Configure, Add, Delete File Systems. Knowledge of Volume management tools like Veritas (now Symantec), Solaris ZFS, LVM.

In larger organizations, some of the tasks above may be divided among different system administrators or members of different organizational groups. For example, a dedicated individual(s) may apply all system upgrades, a Quality Assurance (QA) team may perform testing and validation, and one or more technical writers may be responsible for all technical documentation written for a company. System administrators, in larger organizations, tend not to be systems architects, system engineers, or system designers.

In smaller organizations, the system administrator might also act as technical support, Database Administrator, Network Administrator, Storage (SAN) Administrator or application analyst.

RESULT: studied about the role of system administrator

<u>EXPERIMENT 3</u>

<u>AIM</u>: To familiarize the linux filesystem hierarchy

<u>DESCRIPTION:</u> The Filesystem Hierarchy Standard (FHS) defines the directory structure and directory contents in <u>Unix</u> and <u>Unix-like</u> operating systems, maintained by the <u>Linux Foundation</u>.

Some distributions which generally follow the standard deviate from it in some areas. Common deviations include:

- Modern Linux distributions include a /sys directory as a <u>virtual filesystem</u> (<u>sysfs</u>, comparable to /proc, which is a <u>procfs</u>), which stores and allows modification of the devices connected to the system, whereas many traditional UNIX and Unix-like operating systems use /sys as a <u>symbolic link</u> to the <u>kernel</u> source tree.[citation needed]
- Modern Linux distributions include a /run directory as a <u>temporary filesystem</u> (<u>tmpfs</u>) which stores volatile runtime data, and which is being considered for the next version of the FHS.[6]According to the FHS version 2.3, this data should be stored in /var/run but this was a problem in some cases because this directory isn't always available at early boot. As a result, these programs have had to resort to such trickery as using /dev/.udev, /dev/.mdadm, /dev/.systemd or /dev/.mount directories, even though the device directory isn't intended for such data.[7] Among other advantages, this makes the system easier to use normally with the root filesystem mounted read-only.
  - This is a detailed example from <u>Debian</u>:[8]
    - /dev/.* ⮕ /run/*
    - /dev/shm ⮕ /run/shm
    - /dev/shm/* ⮕ /run/*
    - /etc/* (writeable files) ⮕ /run/*
    - /lib/init/rw ⮕ /run
    - /var/lock ⮕ /run/lock
    - /var/run ⮕ /run
    - /tmp ⮕ /run/tmp
- Many modern <u>UNIX</u> systems[which?] install third party packages into /usr/local while keeping locally developed code in /usr.
- Some Linux distributions no longer differentiate between /lib versus /usr/lib and have /lib symlinked to /usr/lib.
- Some Linux distributions no longer differentiate between /bin versus /usr/bin and /sbin versus /usr/sbin. They symlink /bin to /usr/bin and /sbin to /usr/sbin. And/usr/sbin may get symlinked to /usr/bin.

In the FHS all files and directories appear under the root directory "/", even if they are stored on different physical or virtual devices. Note however that some of these directories may or may not be present on a Unix system depending on whether certain subsystems, such as the X Window System, are installed.

The majority of these directories exist in all UNIX operating systems and are generally used in much the same way; however, the descriptions here are those used specifically for the FHS, and are not considered authoritative for platforms other than Linux.

RESULT: familiarized with linux file system hierarchy

EXPERIMENT 4

AIM: To get familiarized with man pages in linux

DESCRIPTION: A man page (short for **manual page**) is a form of online [software documentation](#) usually found on a [Unix](#) or [Unix-like](#) [operating system](#). Topics covered include [computer programs](#) (including [library](#) and [system calls](#)), formal standards and conventions, and even abstract concepts. A [user](#) may invoke a man page by issuing the man [command](#).

USAGE:

To read a manual page for a Unix command, one can use
man <command_name>

at a [shell](#) prompt: for example, "man ftp". In order to simplify navigation through the output, man generally uses the [less](#) [terminal pager](#).

Pages are traditionally referred to using the notation "name(section)": for example, [ftp(1)](#). The same page name may appear in more than one section of the manual, such as when the names of [system calls](#), user [commands](#), or [macro packages](#) coincide. Examples are [man(1)](#) and [man(7)](#), or [exit(2)](#) and [exit(3)](#).

The syntax for accessing the non-default manual section varies between different man implementations. On Solaris, for example, the syntax for reading [printf(3)](#) is:
man -s 3c printf

On Linux and BSD derivatives the same invocation would be:
man 3 printf

which searches for [printf](#) in section 3 of the man pages.

LAYOUT: All man pages follow a common layout that is optimized for presentation on a simple ASCII text display, possibly without any form of highlighting or font control. Sections present may include:
NAME:The name of the command or function, followed by a one-line description of what it does.
SYNOPSIS:In the case of a command, a formal description of how to run it and what command line options it takes. For program functions, a list of the parameters the function takes and which header file contains its definition. For experienced users, this may be all the documentation needed.
DESCRIPTION:A textual description of the functioning of the command or function.
EXAMPLES:Some examples of common usage.
SEE ALSO:A list of related commands or functions.Other sections may be present, but these are not well standardized across man pages. Common examples include:
OPTIONS, EXIT STATUS, ENVIRONMENT, BUGS, FILES, AUTHOR, REPORTING BUGS, HISTORY and COPYRIGHT.

RESULT: studied about linux man pages

EXPERIMENT 5

AIM: To familiarize with linux shells

DESCRIPTION: Simply put, the shell is a program that takes your commands from the keyboard and gives them to the operating system to perform. In the old days, it was the only user interface available on a Unix computer. Nowadays, we have graphical user interfaces (GUIs) in addition to command line interfaces (CLIs) such as the shell.

Common shells:

sh
The Bourne shell, called "sh," is one of the original shells, developed for Unix computers by Stephen Bourne at AT&T's Bell Labs in 1977. Its long history of use means many software developers are familiar with it. It offers features such as input and output redirection, shell scripting with string and integer variables, and condition testing and looping.
bash
The popularity of sh motivated programmers to develop a shell that was compatible with it, but with several enhancements. Linux systems still offer the sh shell, but "bash" -- the "Bourne-again Shell," based on sh -- has become the new default standard. One attractive feature of bash is its ability to run sh shell scripts unchanged. Shell scripts are complex sets of commands that automate programming and maintenance chores; being able to reuse these scripts saves programmers time. Conveniences not present with the original Bourne shell include command completion and a command history.
csh and tcsh
Developers have written large parts of the Linux operating system in the C and C++ languages. Using C syntax as a model, Bill Joy at Berkeley University developed the "C-shell," csh, in 1978. Ken Greer, working at Carnegie-Mellon University, took csh concepts a step forward with a new shell, tcsh, which Linux systems now offer. Tcsh fixed problems in csh and added command completion, in which the shell makes educated "guesses" as you type, based on your system's directory structure and files. Tcsh does not run bash scripts, as the two have substantial differences.
ksh
David Korn developed the Korn shell, or ksh, about the time tcsh was introduced. Ksh is compatible with sh and bash. Ksh improves on the Bourne shell by adding floating-point arithmetic, job control, command aliasing and command completion. AT&T held proprietary rights to ksh until 2000, when it became open source.

BASIC COMMANDS:
- cp - copy files and directories
- mv - move or rename files and directories
- rm - remove files and directories
- mkdir - create directories
- chmod - modify file access rights
- su - temporarily become the superuser
- chown - change file ownership
- chgrp - change a file's group ownership
- ps - list the processes running on the system

- <u>kill</u> - send a signal to one or more processes (usually to "kill" a process)
- <u>jobs</u> - an alternate way of listing your own processes
- <u>bg</u> - put a process in the background
- <u>fg</u> - put a process in the forground

<u>RESULT:</u> studied about linux shells

PROGRAMS:

1. Perform the following operations in the terminal.
   (a) redirect the list of the files in your home directory starting with s to the file "redirect.txt" and display them in the terminal
   (b) find the number of users logged in to your system
   (c) display the user id, process id, parent process.. of all processes running in your system
2. write a shell script for the following
   (a) to display all the text file containing a digit in it
   (b) to display number of characters, lines, words in a textfile
   (c) to concatenate the contents of two files.
3. Write the shell script for the following.
   (a) To read a student name, course & register no. and print them
   (b) To find the sum, average and product of 4 numbers entered as input.
   (c) Exchange the values of two variables
   (d) To add, subtract, multiply 2 given numbers passed as command line arguments.
   (e) To find average of two numbers given via command line argument.
   (f) To read operands and operators via command line argument and perform the operation.
4. Write the shell script for the following.
   (a) To find largest among three numbers.
   (b) To check whether a given number is even or odd.
   (c) To check whether a given year is leap year or not
   (d) To print the numbers between two given limits.
5. Write the shell script for the following.
   (a) To display the digits which are in odd position of a given 5 digit number
   (b) To find the reverse of a5 digit number
   (c) To find the sum of all the digits of a given number
   (d) To find the occurrences of a particular digit in a given number.