Module - 1

Linux Administration

History of Linux

It is a unix like System

Linux have set of commands to freely interact with the System

The Role of Linux Administrator -

⇓

Common Administration task

Identification and configuration of linux files

Knowledge in Shell programming is the backbone of linux commands

Popular linux distributions :- Fedora, Redhat Enterprise, Debian, Mandriva.

Linux Features

1. Multitasking - Several programs can run in one time
2. Multiuser - Several users on the same machine at same time.
3. Multiplatform - Runs on many different CPU's (not just in Intel)
4. Multiprocessor - SMP support Available on Intel and SPARK platform
5. Multithreading - has native kernal support for multiple independent threads of control within single process memory space.

Linux Advantages

1. Low cost - No need to spend time and money most of the linux softwares come with GNU General Public Licences.
2. Stability : No need of reboot periodically it doesn't to maintain performance
3. Performance: Provide high performance of workstations and on Networks
4. Network

- Network friendliness: Client and server systems are can be set up on any ~~comp~~ linux running computer. It can perform tasks such as network backups faster and more reliable than any other systems.
- Flexibility: Linux can be used in High-end Server Application, Desktop Application and also in embeded system
- Compatibility. Can process all file formats, runs all common Unix Packages
- Choice : Large number of Linux distributions are available.
- Fast and easy. Installation
- Multi-tasking : do many things at one time without slowdown
- Security : Most secured o.s "walls" and flexible file acess permission Prevent unwanted acess

Multiuser - System.

⇩

allows different users to acess resources simultaneously. problem of one user doesn't affect the community

- Time Sharing system, Most batch processing Systems for mainframe computers

The free Software definition.

The freedom to run the program, for any Purpose

Freedom to study it and modify it

The freedom to redistribute it (orginal copy) & redistribute the modified copy.

Open Source Definition

- Source code is
- Most of the free software is always open source.
- Open Source be free but software may not

- Network friendliness: Client and server systems are can be set up on any Linux running computer. It can perform tasks such as network backups faster and more reliable than any other Systems.
- Flexibility: Linux can be used in High-end Server Application, Desktop Application and also in embedded systems
- Compactibility. Can posses all file formats, install common Unix Packages
- Choice : Large number of Linux distributions are available.
- Fast and easy. Installation
- Multitasking : do many things at one time without slowdown
- Security : Most secured OS "walls" and flexible file access permission Prevent unwanted acess.

Multiuser - System.

⇩

allows different users to acess resources simultaneously. problem of one user doesn't affect the community

- Time Sharing system, Most batch processing systems for mainframe computers

The free Software definition.

The freedom to run the program, for any Purpose

Freedom to study it and modify it

The freedom to redistribute it (orginal copy) & redistribute the modified copy.

Open Source Definition

- Source code is freely available
- Most of the open source softwares are free, free software is always free
- Open Source softwares source code can be free but software may not

Intro

Components of linux.

it uses a monolitic kernal, the linux kernal handles -process control, peripheral and file systems acess

The GNU Userland is a important part of linux based system providing the common implementation of the C-Library

The GUI is built on the top of x windows System.

Some components of installed linux are :-

- A bootloader :- This program executed by the computer when it is first turned on and loads the linux kernal into memory (GRUB and LILO)
- Init :- This process launched by linux kernal, and it is the root of the Process tree all processes launched through init. (its shows system service, login prompt)
- Software libraries :- contain code which can be used by running Processes eg: GNU C library
- User Interface :- The User UI also known a shell. (GUI or CLI) or through consol attached through the hardware (which common for embeded system)

Provides Simple interprocess communication.

Types of Users in Linux

linux provides 5 type of ~~user~~ users. according to assessibility

1. Super User or Root User :- Act as administrator of the System Controls acess of other users, permission to do any alterations.
2. System Root User :- It is created by system default. Similar to normal users but have more previllages to acess secure program. that normal user.

3. Normal user: These users are created by root and has limited acess to resources and need permission from root to acess other resources. than allowed.

4. Network user: this type of accounts are used to check network activities. Especially network engineers and Administrators use this account for do network activities

5. Pseudo user: It is completely similar as root user but the root user creates a Pseudo user.

UID or unique identity is given to every user Zero is for root user

upto 500 - System users

500+ to and upto 6000 - Normal user

6000 + Network users.

File Types in Linux

UNIX have - 3 Types

- Regular Files
- Directory files
- Special Files

Special type have 5 sub Types

7 File types In Linux

(ls - to list files)

Special files

- Block files(b)
- Character device file (c)
- pipe file (p)
- Symbolic files
- Socket files

Regular files (ls-l)

(touch) command to create them

It maybe readable file, binary file, Image file or a compressed file.

Directory file (mkdir) for creating

This contain name and location of files stored on a physical device.

Character device file.

Provide only a serial stream of input or output

Block files

These are hardware files most are present in /dev

(disk) command or Partitions create virtual

Pipe files. (Fifo method) first in first out

(make fifo) command bits going in order will be there when coming out.

Symbolic files

These are linked files to other files ln command to create them.

- Socket files

A Socket file is used to pass information blw applications for communication

Purpose socket() is used here.

You can refer socket file using sockfd, we can use read(), write() to read and write() from the system.

Basic Linux System

Linux is multiuser and multitasking

each user have a login name and a password, and each system have a Host name assigned to it.

Creating an account

Newly installed system must have a user account to use the linux

for creating it log in as a root and use : user add or adduser command

$useradd user-name

then create a password too

Shells and COMMANDS

we talk to the system using shell

'cp' command is used to copy.

cp first sample

The content of the first copied into sample.

Changing Password

The command passwd prompts you for your old password

passwd

current password - - -

New password - - -

Retype new.

Files directories and pathnames

A file is a collection of data in the disk. It can be manipulated using its name.

A directory act as folder for other files.
It can contain other directories => called parent directory

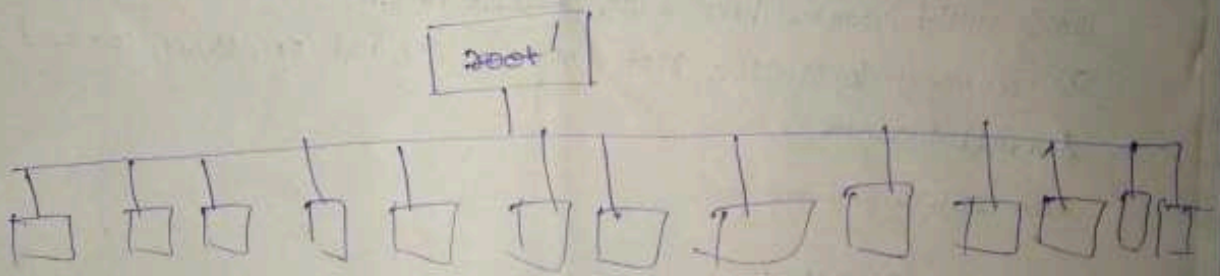
Using name we can identify it.
We can find a file in 2 ways → Absolute path – full path
→ Relative path - path from where you are

Directory Structure

All data in the linux is organized into directories files. All files are organized into directories. These directories are organized into a tree like structure. called directory tree.

directorytree includes directory and all of its files, including content of its sub directories

Root directory is the base of tree hierarchy



1) /--Root
- Every single directory starts from root directory
- Only the root user (Administrator) has "write" privelage under this directory
- /root is root user's home directory which is not same for /

2) /bin --user binaries
- contains binary executable files
- common linux commands you need to use in single user modes are located under this directory
- commands used by all the users of the system are located here.

3 /boot - boot loader files

- Contains boot loader related files
- Kernal initrd, vmlinuz, grub files are located under /boot.

4 /dev - device files

- Contains device files
- These includes terminal devices, usb, or any attached device attached to the system

5 /etc configuration files

Contains configuration files required by all programs.

This contain startup and shutdown shell scripts used to start / activate stop individual programs.

6 /Home - Home directory

Home directories for all users to store their personal files

7 /mnt - Mount directory

- Temporary Mount directory where sysadmins can mount file systems.

8 /lib -- System Libraries

- Contains library files that support the binaries under /bin and /sbin
- Library filenames are ld*, lib*

9. /proc -- process information

contains information about the system process

This is a pseudo file system contains information about running Process

This is a virtual file system with the text information about system resources

10 /root - Home directory for root user

11. /sbin - System binaries

Its like bin contain binary executables - but it contains system administrator, for system maintenance purpose.

12. /tmp - temporary files

- Directory that contains temporary files created by system and users it is temporary deleted when ~~directory are restarted~~, System ~~rebooted~~

13. /usr - User programs

- Contains binaries | libraries | documentation and source code for Second level programs.

/usr | bin => binary files for program

/usr | sbin => System administration

/lib => ~~binaries~~ libraries for sbin

/usr | local - User program that you install from source

14) /var - Variable files

Var - stands for variable files

content of these files that are expected to grow can be found under this directory

This include - System log files

The Current Working directory - PWD - shows current directory

ls - used to list the directories and files

Moving around the directory

cd - change directory cd bca → mkdir - make directory

cd .. reverse change directory bca/ - - - cp - copy

cd → ← cd .. mv - move

rm - remove

Types of Shells

A Shell refers to a program that is used to interpret the typed commands the user sends to the OS. A Shell is not only an interpreter for the interactive commands you type at the prompt but also a powerful programming language.

Sh - Bourne Shell - Its Orginally developed for Unix System by Stephen Bourne.

Features - Input output redirection
Shell scripting
Shell scripting with string and integer variables
condition testing and looping

'bash - "Bourne again shell - became new default standard

Features
-> Run sh programs unchanged
-> complex set commands that automate programs and maintenance chores: being able to reuse these scripts saves programmers time.
include command completion and command history

Ksh - David Korn developed the korn Shell, or Ksh
- Tcsh was introduced
- compactable with Sh and bash
- introduced floating-point, job control, command aliasing and command completion.

Getting Online help

manual pages are used - man -ls

Wild cards

The ability to refer to more than one file by name using special characters

The wild card "*" Specifies any character or string of characters in a filename
"?" expands to only a single character

Linux Administration

File Permission

Every file ~~in linux base~~/directory in linux owned by a specific user and group

∴ The file permission seperately ~~g~~ divided into ~~users~~ users, group, others

Total 3 types of users are there.

1. Owner

The owner permission applies to owner of a ~~group~~ file/directory the default user became the owner of the group.

2. group

The group permission apply to a group of user that ~~assigned~~ to the file or directory.

3. Others

This apply to all other users on the system.

Permission Types.

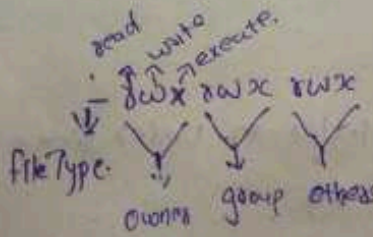
Read - To read the content of the file (refer to users

write - Capability to write or modify a file or directory

Execute - To execute a file or viewer the content of a directory.

Viewing the Permission

Using (ls-l) we can view the file permissions



bourne shell Scalpt

Shell Scalpt = Set of shell commands groupped into a file. and executed by the shell

installed on /bin/sh.

First shell Scalpt

Step 1: text edition

like Vi editor in CLI

Step 2: Enter the commands.

Step 3. Save the file in current working directory.

Step 4: Run the Scalpt.

sh filename.sh.

It gives the output

'who am i' = to display the current ~~working directory~~ username.

'-c' : to stay in same line.

Shell variables

Shell variables is used to store and manipulate informations within the shell program.

It doesn't have datatypes like int, char, string etc..

Rules for making Variables

Variable name = Alphabet, numbers, underScores

no comma or blank space

first character must be ~~numbers or~~ letter or underScore

no numbers is allowed at the beginning

variable name is case sensitive

Two types of variables are there System variables and user variables

System variables = have Special meaning

using # $ infront of variables we can access it.

Common environmental variables. System Variables

Home : The path to home directory

IFS : Internal Field Seperator Character
(space, tab, newline)

- Path : Colon seperator list of directories to search for the command to execute

PWD : Current Working directory

USER : The name of the user running current session.

- PS1, PS2 : The strings to use the primary and Secondary prompt.

Vi-editor

Vi is the one of popular text editor under UNIX type

Vi is an editor which is fully text model
All the actions carried out with the help of commands.

Vi _ filename.sh

when the file open we can move the cursors using H, J, K, L

Vi modes

Vi has 3 modes (command mode or Regular mode), insert mode, last line mode.

Command line / Regular mode

When we start the Vi editor it is in the command mode we use commands to edit the files. or change to other mode

example /t -> moves the cursor position to the left.

insert mode

- For inserting the insertion of new text, editing of existing text or replacement of text carried out in Insert mode

to move from command mode to insertion press I to return from Insert mode to command mode press 'esc'

last line mode: This mode permit us to give commands at the command line at the bottom of the Vi screen called the command line

Any command begins with : colon is last line mode (ex command mode)

:q quit without saving
:wq save the document and quit the editor
:filename Save the document ~~and~~ under ~~the~~ a specified name.

'Editing ~~to~~ Commands ~~Mode~~' in Command Mode

~~for~~ x - delete a character under a cursor
dd - delete the line under the cursor
dxd - delete x lines starting ~~starting~~ with one under under cursor
nx - delete n characters starting with one under cursor.

1) Conditional constructors.

If

```
If statement:
    if {condition}
        then
            code
    fi
```

If else

```
if condition
then
    {code}
else
    {code}
fi
```

elif

```
if (condition)
then
    code
elif (condi)
then
    code
fi
```

Relational operators

- gt greather than
- lt less than
- ge greather than or equal
- le less than or equal
- ne not equal
- eq equal to.

2) case control structure

example

```
read n
case ~~in A~~ $n in
    1) exp code
    2) exp code
    :
    :
```

While Loop

Execute a set of commands until a particular condition occurs.

```
while [ condion ]
  do {
     code
    }
  done.
```

for loop

```
for { control variable in Value 1, Value 2 ... Value n.
  do
     code
  done.
```

example

```
for a in 1,2,3,4
  do
    echo "Hai"
  done.
```

Arithmetic in Shell Script:

In Shell programming every ~~everything~~ is considered as string wat if we want to arithmetic operation on them use `expr` - for evaluating arithmetic operations

File permissions

Permission weightage

read	4
write	2
execute	1
Total	7

rwx=7

when all users have all permissions then 7 7 7

Changing file Permissions

The existing file permission one can be change by the owner or by the user super user.

It is done by using chmod.

```
chmod 700 myfile
```

There is another syntax for chmod

```
chmod [who] [action] [Permission] [filename]
```

who- to whom the permission are to be assigned.

u-owner, g-group, a - all

action { + for add permission
- Remove
= add specified permission and remove others.

example chw my file

Directory Permission

A directory is a file that contains names of the file or directory of that directory

Read permission - allows the listing of directory content
write permission - Per allows to create or remove files in it
Execute Permission - User can pass through the directory in search of sub directory

Masking file Permissions

Linux assign default file permission when you create a file.
this permission is assigned on the value stored in an environment variable called 'Umask'

Umask - User file creation mask
the term implies which permission to be mask or hide.
The value of umask can be get by $ Umask.
The result will be in octal format
The first 3 digits shows the file permissions that created
1st digit - owner, 2nd digit - group 3rd - others

Mounting a file System

- Attach a file device to file system hierarchy
- Mounting a file system attach that file system to a directory called mountpoint and make it available to system.

Root / -> To and from root file systems can be connected or removed

Methods
Methods
manually with mount command, Automatically when booted. Automatically when file system is accessed, Shut down to unmount, manually by unmount command.