Ubuntu

Install vm

<https://www.virtualbox.org/wiki/Downloads>

Advanced Packaging Tool (APT)

https://ubuntu.com/download/desktop/thank-you?version=20.04.4&architecture=amd64

intro to unix:-

Facility of interconnecting commands through pipes and filters

permit the user to create complex programs from simple programs

Facility of background processing helps the user in effective

utilization of time

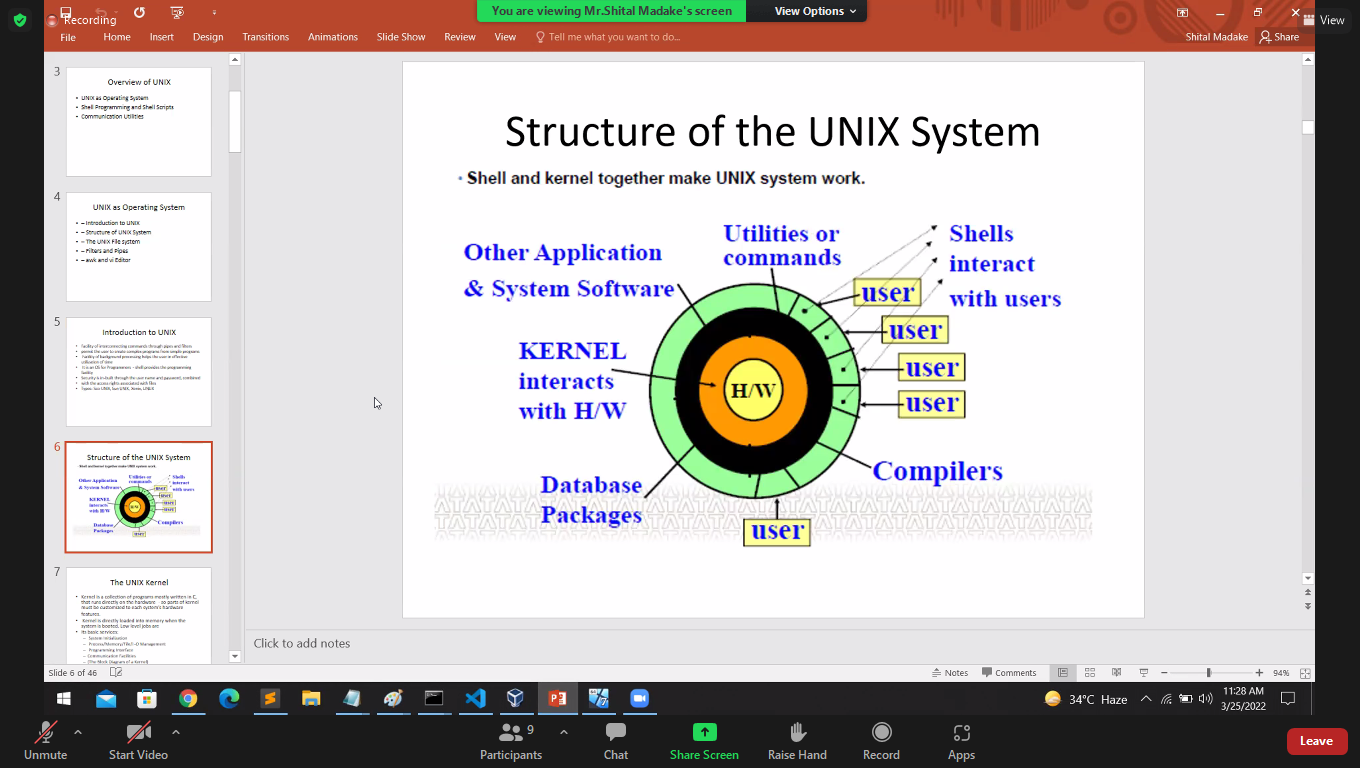
It is an OS for Programmers - shell provides the programming

facility

Security is in-built through the user name and password, combined

with the access rights associated with files

Types: Sco UNIX, Sun UNIX, Xenix, LINUX



The unix kernel system

Kernel is a collection of programs mostly written in C, that runs directly on the hardware - so parts of kernel must be customized to each system’s hardware

features.

Kernel is directly loaded into memory when the system is booted. Low level jobs are

its basic services:

System Initialization

Process/Memory/File/I-O Management

Programming Interface

Communication Facilities

(The Block Diagram of a Kernel)

Kernel process

The kernel process is the first (root) process that comes into existence when the system is booted. Its process\_id and group\_id are both 0.

In establishing a multi-programming environment, the kernel process then creates the init

process with its process\_id of 1 and group\_id, 0, showing that process 1 is part of process 0.

The init process creates and manages terminal processes for active terminals in the system.

At the time of creation, each terminal process belongs to process group 0 and is made to execute a program called getty. Once each terminal process (now called a getty process)

establishes communication with the associated terminal, it displays a login message and

waits for a user to input a user name (Process group of the init process).

When a user starts a dialogue, the getty process receives the user name and leaves the task of validating the user’s password to another program called login. The same terminal process is now called a login process.

The login process, after validating the user’s password, calls a command line program, the login shell to run in the same process. The same terminal process is now called a shell process.

Each shell process now establishes a new process group and becomes ready to process the user commands. A shell process is the initiating process by which each terminal maintains the user session.

While interpreting a command, the shell creates an execution thread and then assigns the requested command program to this new process.

Both the shell and the new process proceed independently in separate execution threads. The parent shell process normally waits until child process completes its execution.

Commands

Ps

Env detail

Cntr+l clear

Pwd – present working dir

Ls –al – all list

Ls-r – directory

Ls –a -hidden directory

Cd /- change path

Vi file – text in file

Cat file- run file

Mkdir mydir- make file

rkdir mydir-remove file

cp myfile mydir- copy file

Command Function

pwd print working directory

cd change directory

mkdir make directory

rmdir remove directory

ls list contents of directory

cat display contents of files

cp copy one file to another

rm remove files

mv change name of a file/directory

or move file from one directory to another

date display system date

tput clear clear screen

who display logged-in users’ names

echo display string; used for prompt

ln create another link to a file

Shell includes programming language features which can be used to build shell scripts for performing complex operations.

Shell scripts are frequently used sequence of shell commands stored in a file; file-name can be used later to execute the stored sequence.

Types of Shells –

Bourne Shell - original command interpreter developed at AT&T by Stephen R. Bourne; fastest official shell distributed with UNIX systems (executable filename sh)

C Shell - developed by William Joy and others at UCB; gets its name from C due to syntax resemblance of its programming language (executable file name csh)

Korn Shell - developed by David Korn, combines best features of both shells, not popular (executable file name ksh)

Restricted Shell - restricted version of Bourne shell, typically used for guest logins and in secure installations (executable file name).

UNIX utilities or commands are collection of about 200 programs (ed, sed, awk, vi, grep, make, link, debug, etc.) that service day to- day processing requirements. They are invoked through the shell, which is itself another utility.

More than a 1000 UNIX-based application programs like DBMS,word processors, language processors, accounting software, etc. are available from independent vendors

1. Find out the directory where you are currently placed.

2. Create a directory structure as follows under your home directory.

HRD OFFICE

HR HRAG TRNG FIN MFG

3. From your home directory list the directory structure that is under the root.

4. Create 3 empty files empty1, empty2 and empty3 under your home directory.

5. Create a file text under HR and type a few lines. Display the contents of text.

6. Make a copy of text into another file copytext under TRNG.

7. Create a file matter under HRD and input a few sentences for it.

8. Combine the contents of the files text and matter into the file empty1.

9. Compare text and copytext, matter and empty1 using cmp and comm.

10. Create a link tmp under FIN for matter. Get inode numbers for both files?

11. Change the permissions for copytext to “rwxrw\_rw\_”.

Links:-

symlink is a symbolic linux/UNIX link

two form

1] Soft link : which are similar like shortcuts

and they can point to file or directory

we can use for any file system

2] Hard link:- they are also similar to shortcuts

cannot be created for a folder or file in a dirrent file system.

ln -s <path to file/folder> to\_be\_linked

ln file1 file2:- link both file if edit 1 reflect other also

wc file name:- line words characters of that file

/home – It holds user’s home directories. In other UNIX systems, this can be /usr directory.

/bin – It holds many of the basic Linux programs; bin stands for binaries, files that are executable.

/usr – It holds many user-oriented directories:

bin – It holds user-oriented Linux programs.

sbin – It holds system administration files.

spool – It has several subdirectories:

. mail holds mail files

. spool holds files to be printed

. uucp holds files copied between Linux machines.

docs – various documents including useful Linux info

man – man pages accessed by typing the man <command>

games – the fun stuff!

/sbin – It holds system files that are usually run automatically.

/etc – It and its subdirectories hold many of Linux config files.

/dev – It holds device files. All info sent to /dev/null is thrown into trash. Your terminal is one of the

/dev/tty files.

Tty –teletypewriter

The UFS resides on a single logical disk. A logical disk is a disk partition comprising of a set of

consecutive cylinders.

• UFS further subdivides a partition into one or more cylinder groups and attempts to allocate

inodes and related data blocks from the same cylinder group, thus minimizing the disk head

movements.

• At the beginning of the logical disk lies the boot block of UNIX operating system containing the

bootstrap program.

• It is followed by repetitive cylinder groups each one containing a super block, cylinder group

block, inode list and the data area.

• Each cylinder group contains a duplicate copy of the super block. The super block contains the

size of file system, number of free blocks, index of next free block in free block list, size of inode

list, number of free inodes, index of next free inode in free inode list.

• The cylinder group block contains a number of inodes and corresponding data blocks for that

cylinder group. The block size is a power of 2 (>=4096).

Read(r)

Write(w)

excute(x)

User owner

Group

other

position character ownership

1 - denotes file type

2-4 rw- permission for user

5-7 rw- permission for group

8-10 r-- permission for other

Lab Session 1

Log into UNIX system as per your user\_id and password.

1. Find out the directory where you are currently placed.

2. Create a directory structure as follows under your home directory.

HRD OFFICE

HR HRAG TRNG FIN MFG

3. From your home directory list the directory structure that is under the root.

4. Create 3 empty files empty1, empty2 and empty3 under your home directory.

5. Create a file text under HR and type a few lines. Display the contents of text.

6. Make a copy of text into another file copytext under TRNG.

7. Create a file matter under HRD and input a few sentences for it.

8. Combine the contents of the files text and matter into the file empty1.

9. Compare text and copytext, matter and empty1 using cmp and comm.

10. Create a link tmp under FIN for matter. Get inode numbers for both files?

11. Change the permissions for copytext to “rwxrw\_rw\_”.

12. Delete the file text. Rename copytext to mytext.

13. Copy mytext and matter under OFFICE directory and then move both files under MFG directory.

14. Copy OFFICE directory structure under HRAG directory in one shot.

15. Identify all empty files created earlier and delete them interactively.

16. List the directory structure under your home directory and then delete the same in one shot.

17. Use vi editor to create a file student under your home directory. Input 12 lines for the name, age, roll number and sex. Use ‘|’ as the field separator. Search some pattern using the editor’s internal commands.

18. (a) Use cut and paste commands to alter the field ordering in student.

(b) Sort student: major key age, minor key sex; store sorted output in std.

19. Get all users currently logged on to your system with column headers for the multi-column output and get the complete pathname of your terminal device.

20. Try the command: $ who | tee file1 | wc –l | tee file2

How to use tee such that the list of the logged users as well as their count is displayed only on the terminal and the same is not stored in any file.