CS35L
Software Construction Laboratory

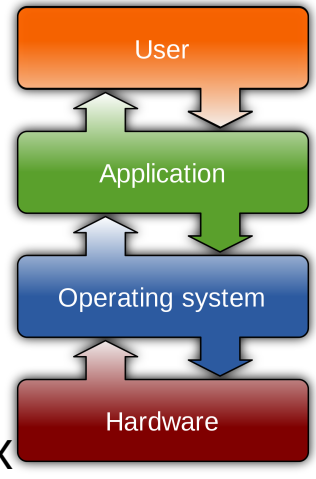
Nandan Parikh

Week 1 : Lecture 1

SPRING 2019

What is an Operating System?

- Most important software that runs on a computer
- Manages memory, processes, other softwares and hardwares
- Makes human to computer communication easy
- Computer is useless without an OS!
- Brief history of Operating Systems: http://www.informit.com/articles/article.aspx?p=24972
- OS Examples : Windows (Windows 10, 8..), MacOS (OS X), UNIX



Source : https://en.wikipedia.org/wiki/Operating_system

Multiuser and Multi-process Operating System

- Multi-User OS- Allow many users to access/work on a single system at the same time (as long as they have their own terminal)
- Multi-Process OS- Allows many processes, programs and applications to run simultaneously.
- Variants :
  - Single User Single Task
  - Single User Multi Task
  - Multi User OS

User Interfaces: CLI v/s GUI

Command Line Interface

- Steep learning curve
- Pure control (e.g., scripting)
- Speed: Only keyboard, faster performance
- Consumes less resources
- No change; less diverse

Graphical User Interface

Intuitive
Limited Control
Mouse + keyboard; Slower
More resources; e.g. loading icons, fonts, etc.
Frequent changes; More diverse

Debian GNU/Linux

- Clone of UNIX
- Linux is just a kernel.
- What is a kernel?
  - Core of any OS
  - Allocates time and memory to programs
  - Interfaces applications with the physical hardware
  - Allows communication between different processes: inter-process communication (IPC)
- Linux distribution make the Linux kernel a completely usable OS by adding various applications
- Linux distribution = GUI + GNU utilities (cp,mv,ls,etc) + installation and management tools + GNU compilers (c/c++) + Editors(vi/emacs) + ….
- Shell : Interface between the user and kernel

Basics of Shell

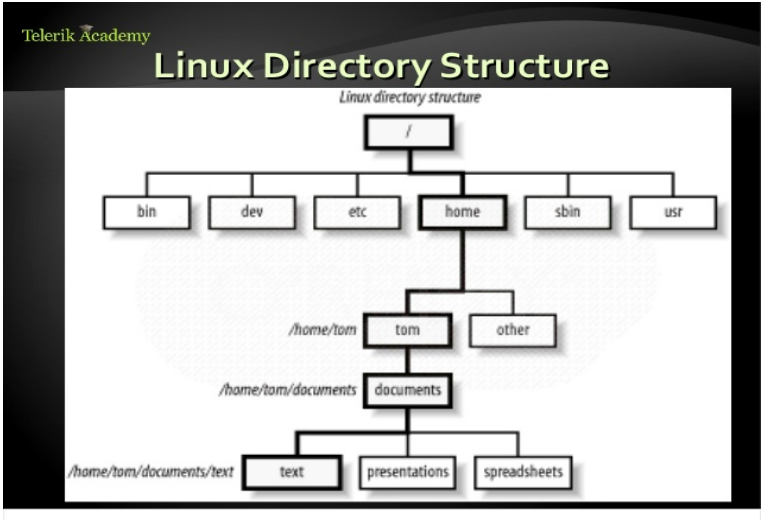
- Outermost layer around the kernel; hence called shell !
- Can be used as CLI as well as GUI depending upon the task/operation
- Examples:
  - CLI shell in Windows :
    - Command Prompt
  - CLI shell in UNIX :
    - Shell
  - CLI in Mac :
    - Terminal
- Basic shell commands:
  - <up arrow>: previous command
  - <tab>: auto-complete
  - !!: replace with previous command
  - ![*str*]: refer to previous command with str

Files and Processes

- Everything is either a process or a file
- Process: an executing program identified by PID
- File: collection of data
    - A document
    - Text of program written in high-level language
    - Executable
    - Directory
    - Devices

Linux File System Layout

- Tree Structure Hierarchy



Source: https://www.slideshare.net/azilian/4-linux-file-systems-18175783

- Only One Root- '/'
- Directories are also files
  - E.g. home, tom
- Regular files can only be leaves
  - E.g. text, spreadsheets, etc

The Basics: Moving Around

- pwd: print working directory
- cd: change directory
  - ~ home directory
  - . current directory
  - / root directory, or directory separator
  - .. parent directory

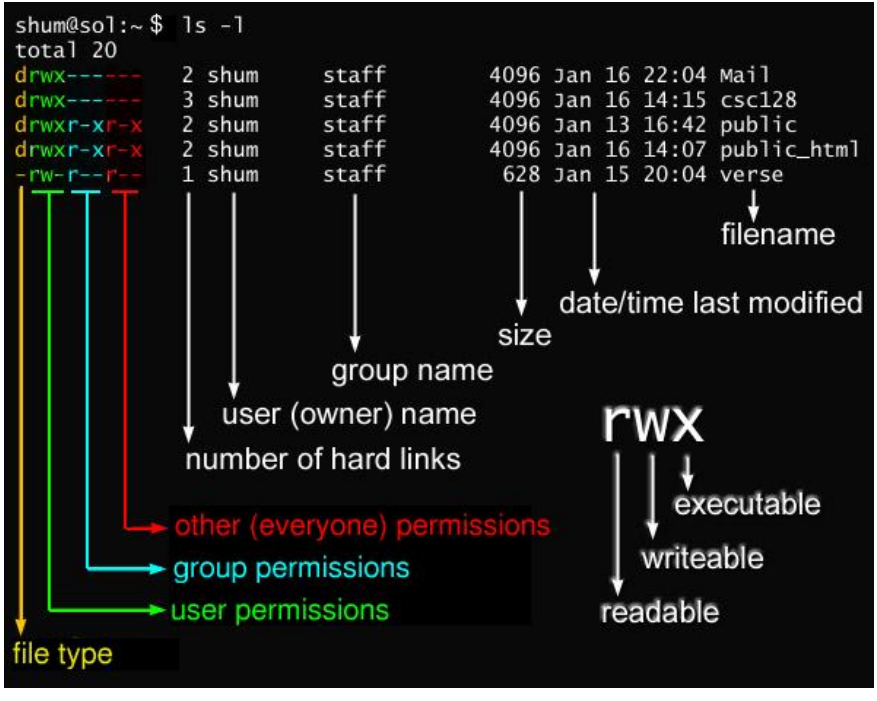
The Basics: Dealing with Files

- mv: move/rename a file
- cp: copy a file
- rm: remove a file
  - r: remove directories and their contents recursively
- mkdir: make a directory
- rmdir: remove an empty directory
- ls: list contents of a directory
  - d: list only directories
  - a: list all files including hidden ones
  - l: show long listing including permission info
  - s: show size of each file, in blocks

The Basics: Changing File Attributes

- ln : creates a link
  - Hard links : Point to physical Data
  - Additional name for an existing file
    - ln file1 hlink1
  - Soft Links/ Symbolic Links (-s): Point to file
    - ln –s <source file> <my file>
- touch: update access & modification time to current time
  - touch filename
  - touch -t 201101311759.30 filename
    - Change filename's access & modification time to (year 2011 January day 31 time 17:59:30)

The Basics: File Permissions



File Permissions

- chmod
  - read (r), write (w), executable (x)
  - User, group, others

Reference	Class	Description
u	user	the owner of the file
g	group	users who are members of the file's group
o	others	users who are not the owner of the file or members of the group
a	all	all three of the above, is the same as *ugo*

chmod contd…

- Numeric

#	Permission
7	full
6	read and write
5	read and execute
4	read only
3	write and execute
2	write only
1	execute only
0	none

Symbolic

Operator	Description
+	adds the specified modes to the specified classes
-	removes the specified modes from the specified classes
=	the modes specified are to be made the exact modes for the specifi ed classes

Mode	Name	Description
r	read	read a file or list a directory's contents
w	write	write to a file or directory
x	execute	execute a file or recurse a directory tree

Special permissions

- setuid : set user ID on execution
- Permits users to run certain programs with escalated privileges
- E.g. : chmod u+s file1
- When an executable file's setuid permission is set, users may access the program with a level of access that matches the owner
- E.g. passwd command

```
ls -l /usr/bin/passwd
```

```
-rwsr-xr-x 1 root 54192 Nov 20 17:03 /usr/bin/passwd
```

Special permissions contd…

- setgid : Grants permission of the group which owns the file
- E.g. : chmod g+s file2

```
ls -l myfile2
```

```
-rw-r-sr-- 1 user 0 Mar 6 10:46 myfile2
```

Basic Shell Commands

- man
- cat
- head
- tail
- du
- ps

kill
diff
cmp
wc
sort
echo

The Basics: Redirection

- > *file*: write stdout to a file
- >> *file*: append stdout to a file
- < *file*: use contents of a file as stdin

find command

- -type: type of a file (e.g: directory, symbolic link)
- -perm: permission of a file
- -name: name of a file
- -user: owner of a file
- -maxdepth: how many levels to search

find contd…

- ?: matches any single character in a filename
- *: matches one or more characters in a filename
- []: matches any one of the characters between the brackets. Use '-' to separate a range of consecutive characters.
- Examples:
  - find . -name my*
  - find . -name my* -type f
  - find / -type f -name myfile

man command

- Extensive documentation that comes preinstalled with almost all substantial Unix and Unix-like operating systems
- Usage
  - read a manual page for a Linux command
    - man <command_name>
- Hit “q” to get out of man page

wh commands

- whatis <command>: returns Name section of man page
- whereis <command>: locates the binary, source, and manual page files for a command
- which <command>: locates only the binary

diff command

- A file comparison utility that outputs the differences between two files.
- Usage:
  - diff file1 file2
  - diff –u file1 file2 (unified format)

wget command

- A computer program that retrieves content from web servers
- Usage
  - wget <URL>