Linux Session Part I

Kesavan M

180976

Overview

- What is an Operating System?
- UNIX History
- Parts of the UNIX OS
- Flavors of UNIX
- Before Linux
- GNU project
- Beginning of Linux
- Linux Today
- Linux - free software
- Disto

- Linux Shell
- Directory Tree
- Important subdirectories
- Home directory
- Basic Commands
- Pattern matching
- Redirection and Pipes
- Vi(m)
- Shell Scripting
- Cheat Sheets
- QA

What is an Operating System?

- The operating system (OS) is the program which starts up when you turn on your computer and runs underneath all other programs - without it nothing would happen at all.

- In simple terms, an operating system is a *manager*. It manages all the available resources on a computer, from the CPU, to memory, to hard disk accesses.

- Tasks the operating system must perform:
  - **Control Hardware** - The operating system controls all the parts of the computer and attempts to get everything working together.
  - **Run Applications** - Another job the OS does is run application software. This would include word processors, web browsers, games, etc...
  - **Manage Data and Files** - The OS makes it easy for you to organize your computer. Through the OS you are able to do a number of things to data, including copy, move, delete, and rename it. This makes it much easier to find and organize what you have.

UNIX History

- The UNIX operating system was born in the late 1960s. It originally began as a one man project led by Ken Thompson of Bell Labs, and has since grown to become the most widely used operating system.

- In the time since UNIX was first developed, it has gone through <u>many</u> different generations and even mutations.

  - Some differ substantially from the original version, like Berkeley Software Distribution (BSD) or Linux.

  - Others, still contain major portions that are based on the original source code.

Parts of the UNIX OS

- **The Kernel** - handles memory management, input and output requests, and program scheduling. Technically speaking, the kernel *is* the OS. It provides the basic software connection to the hardware. The kernel is very complex and deals with the inner workings of these things, and is beyond the scope of this course. The kernel is covered in detail in "Operating Systems."

- **The Shell and Graphical User Interfaces (GUIs)** - basic UNIX shells provides a "command line" interface which allows the user to type in commands. These commands are translated by the shell into something the kernel can comprehend, and then executed by the kernel.

- **The Built-in System Utilities** - are programs that allow a user to perform tasks which involve complex actions. Utilities provide user interface functions that are basic to an operating system, but which are too complex to be built into the shell. Examples of utilities are programs that let us see the contents of a directory, move & copy files, remove files, etc...

- **Application Software & Utilities** – these are not part of the operating system, per se. They are additional programs that are bundled with the OS distribution, or available separately. These can range from additional or different versions of basic utilities, to full scale commercial applications.

Flavors of UNIX

- These can be grouped into two categories: **Open Source** and **Proprietary**
- **Proprietary:** (redistribution and modification prohibited or restricted; not free)
  - Solaris - Access to a Solaris UNIX server (solaris.gl.umbc.edu) via SSH access.
  - IRIX - There are a couple of IRIX machines in the basement of the library, as well as server access (irix.gl.umbc.edu) via SSH access.
  - Mac OS X - no access currently available at UMBC.
  - and many others...
- **Open Source:** (source code is readily available and free to modify)
  - FreeBSD - no access currently available at UMBC.
  - Linux Distributions - access is available in the form of dual-boot PCs scattered throughout the Engineering building. There are also several Linux servers (linux.gl.umbc.edu) through which access is available.
    - RedHat (used by UMBC) and the Fedora Project (maintained by RedHat)
    - Mandrake
    - Debian
    - SuSE
    - Slackware
    - and many others...
- As a side note, Linux is a open source UNIX-based OS that was originally developed in 1991 by Linus Torvalds, a Finnish undergraduate student.

Before Linux

- In 80's, Microsoft's DOS was the dominated OS for PC
- Apple MAC was better, but expensive
- UNIX was much better, but much, much more expensive. Only for minicomputer for commercial applications
- People was looking for a UNIX based system, which is cheaper and can run on PC
- Both DOS, MAC and UNIX were proprietary, i.e., the source code of their kernel is protected
- No modification is possible without paying high license fees

GNU project

- Established in 1984 by Richard Stallman, who believes that software should be free from restrictions against copying or modification in order to make better and efficient computer programs

  –GNU is a recursive acronym for "GNU's Not Unix"

  –Aim at developing a complete Unix-like operating system which is free for copying and modification

  –Companies make their money by maintaining and distributing the software, e.g. optimally packaging the software with different tools (Redhat, Slackware, Mandrake, SuSE, etc)

  –Stallman built the first free GNU C Compiler in 1991. But still, an OS was yet to be developed

Beginning of Linux

- A famous professor Andrew Tanenbaum developed Minix, a simplified version of UNIX that runs on PC

- Minix is for class teaching only. No intention for commercial use

- In Sept 1991, Linus Torvalds, a second year student of Computer Science at the University of Helsinki, developed the preliminary kernel of Linux, known as Linux version 0.0.1

Linux Today

- Linux has been used for many computing platforms
  - PC, PDA, Supercomputer,…
- Not only character user interface but graphical user interface is available
- Commercial vendors moved in Linux itself to provide freely distributed code. They make their money by compiling up various software and gathering them in a distributable format
 - Red Hat, Slackware, etc

Linux - free software

• Free software, as defined by the FSF (Free Software Foundation), is a "matter of liberty, not price." To qualify as free software by FSF standards, you must be able to:

- Run the program for any purpose you want to, rather than be restricted in what you can use it for.
- View the program's source code.
- Study the program's source code and modify it if you need to.
- Share the program with others.
- Improve the program and release those improvements so that others can use them.

Disto

- Red Hat Linux : One of the original Linux distribution.
- The commercial, nonfree version is Red Hat Enterprise Linux, which is aimed at big companies using Linux servers and desktops in a big way. (NJIT)
- Free version: Fedora Project.

- Debian GNU/Linux : A free software distribution. Popular for use on servers. However, Debian is not what many would consider a distribution for beginners, as it's not designed with ease of use in mind.

- SuSE Linux : SuSE was recently purchased by Novell. This distribution is primarily available for pay because it contains many commercial programs, although there's a stripped-down free version that you can download.

- Mandrake Linux : Mandrake is perhaps strongest on the desktop. Originally based off of Red Hat Linux.

- Gentoo Linux : Gentoo is a specialty distribution meant for programmers.

Linux Shell

- Shell interprets the command and request service from kernel
- Similar to DOS but DOS has only one set of interface while Linux can select different shell
  - Bourne Again shell (Bash), TC shell (Tcsh), Z shell (Zsh)

- Different shell has similar but different functionality
- Bash is the default for Linux
- Graphical user interface of Linux is in fact an application program work on the shell

whoami

ls

pwd

Bash, Tcsh, Zsh

Kernel
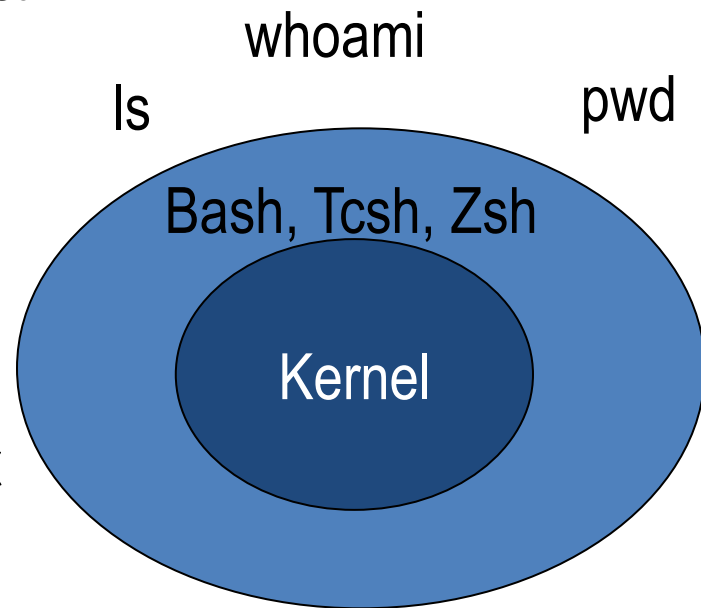
Directory Tree

- When you log on the the Linux OS using your username you are automatically located in your home directory.
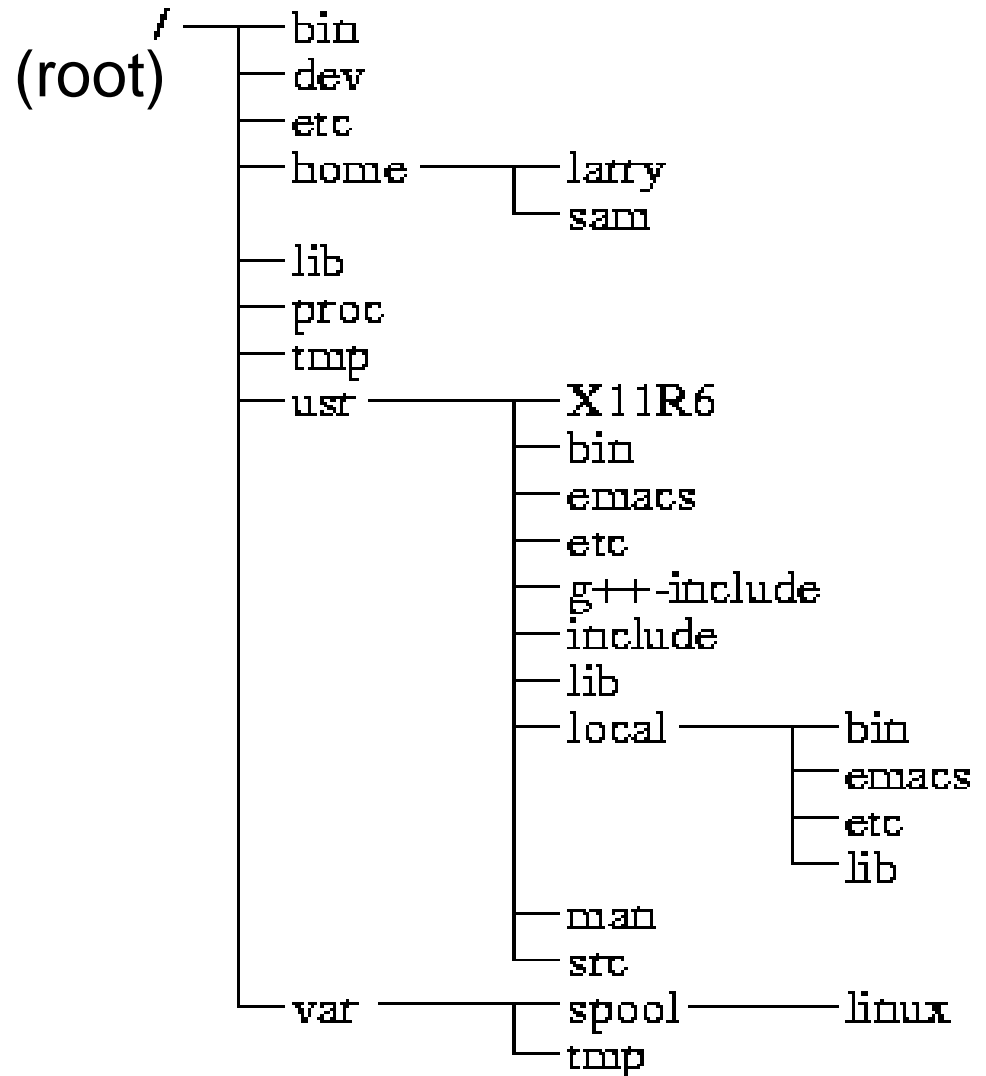
Important subdirectories

- **/bin** : Important Linux commands available to the average user.
- **/boot** : The files necessary for the system to boot. Not all Linux distributions use this one. Fedora does.
- **/dev** : All device drivers. Device drivers are the files that your Linux system uses to talk to your hardware. For example, there's a file in the /dev directory for your particular make and model of monitor, and all of your Linux computer's communications with the monitor go through that file.
- **/etc** : System configuration files.
- **/home** : Every user except root gets her own folder in here, named for her login account. So, the user who logs in with linda has the directory /home/linda, where all of her personal files are kept.
- **/lib** : System libraries. Libraries are just bunches of programming code that the programs on your system use to get things done.

Important subdirectories

- **/mnt :** Mount points. When you temporarily load the contents of a CD-ROM or USB drive, you typically use a special name under /mnt. For example, many distributions (including Fedora) come, by default, with the directory /mnt/cdrom, which is where your CD-ROM drive's contents are made accessible.
- **/root** : The root user's home directory.
- **/sbin** : Essential commands that are only for the system administrator.
- **/tmp** : Temporary files and storage space. Don't put anything in here that you want to keep. Most Linux distributions (including Fedora) are set up to delete any file that's been in this directory longer than three days.
- **/usr** : Programs and data that can be shared across many systems and don't need to be changed.
- **/var** : Data that changes constantly (log files that contain information about what's happening on your system, data on its way to the printer, and so on).

Home directory

- You can see what your home directory is called by entering

- pwd (print current working directory)

Basic Commands

- ls
  - $ ls -l
  - $ ls -a
  - $ ls -la
  - $ ls -l --sort=time
  - $ ls -l --sort=size -r
- cd
  - $ cd /usr/bin
- pwd
  - $ pwd
- ~
  - $ cd ~
- *~user*
  - $ cd ~weesan
- What will "cd ~/weesan" do?

which
  $ which ls
whereis
  $ whereis ls
locate
  $ locate stdio.h
  $ locate iostream
rpm
  $ rpm -q bash
  $ rpm -qa
  $ rpm -qa | sort | less
find
  $ find / | grep stdio.h
  $ find /usr/include | grep stdio.h

Basic Commands

- echo
  - $ echo "Hello World"
  - $ echo -n "Hello World"
- cat
  - $ cat /etc/motd
  - $ cat /proc/cpuinfo
- cp
  - $ cp foo bar
  - $ cp -a foo bar
- mv
  - $ mv foo bar
- mkdir
  - $ mkdir foo

rm
  $ rm foo
  $ rm -rf foo
  $ rm -i foo
  $ rm -- -foo
chgrp
  $ chgrp bar /home/foo
chsh
  $ chsh foo
chfn
  $ chfn foo
chown
  $ chown -R foo:bar /home/foo

Basic Commands

- tar
  - $ tar cvfp lab1.tar lab1
- gzip
  - $ gzip -9 lab1.tar
- untar & ungzip
  - $ gzip -cd lab1.tar.gz | tar xvf –
  - $ tar xvfz lab1.tar.gz
- touch
  - $ touch foo
  - $ cat /dev/null > foo

Basic Commands

- Pipe
  - $ cal > foo
  - $ cat /dev/zero > foo
  - $ cat < /etc/passwd
  - $ who | cut -d' ' -f1 | sort | uniq | wc –l
- backtick
  - $ echo "The date is `date`"
  - $ echo `seq 1 10`
- Hard, soft (symbolic) link
  - ln vmlinuz-2.6.24.4 vmlinuz
  - ln -s firefox-2.0.0.3 firefox

Basic Commands

- Disk usage
  - $ df -h /
- File space usage
  - $ du -sxh ~/
- **Advance stuff** ☺
  - $ ssh eon who
  - $ ssh eon 'cd .html ; tar cvfp - cs183 | gzip -9c' | tar xvfpz -
  - $ ssh kilo-1 'tar cvfp - /extra/weesan' | tar xvfp - -C /

Pattern matching

- grep – GNU Regular Expression Processor
- Finds the words / patterns matching with the search and displays the line containing the patterns.
- Search is limited to a file

Redirection and Pipes

- Redirection
  - Input redirection
    - wc < file1 – Content of file 1 is given as input for wc command that counts the no of lines, words and characters in a file
  - Output redirection
    - cat file > newfile – Copies file's content to newfile. Over writes the existing content
    - cat file >> newfile – Appends the new content to the existing content
- Pipes
  - Output of first command is input for the second and so on
  - who | wc –l – Number of lines in the output of who command will be displayed

Vi(m)

- 2 modes
  - Input mode
    - ESC to back to cmd mode
  - Command mode
    - Cursor movement
      - h (left), j (down), k (up), l (right)
      - ^f (page down)
      - ^b (page up)
      - ^ (first char.)
      - $ (last char.)
      - G (bottom page)
      - :1 (goto first line)
    - Swtch to input mode
      - a (append)
      - i (insert)
      - o (insert line after
      - O (insert line before)

- Delete
  - dd (delete a line)
  - d10d (delete 10 lines)
  - d$ (delete till end of line)
  - dG (delete till end of file)
  - x (current char.)
- Paste
  - p (paste after)
  - P (paste before)
- Undo
  - u
- Search
  - /
- Save/Quit
  - :w (write)
  - :q (quit)
  - :wq (write and quit)
  - :q! (give up changes)

Shell Scripting

- Shell scripting is the most useful and powerful feature in Linux
  - Minimizes typing of repetitive command
  - Can schedule jobs to run in the system
  - Can initiate back up activities for system administration
  - Similar to batch files in DOS, but more powerful than Batch files

fwunixref.pdf

ux4beginner.p

n_cheat_sheet

Page Linux M

Page Linux M

x-cheat-sheet

QA ?