Linux Basics & Internals

Team Emertxe
Day 1



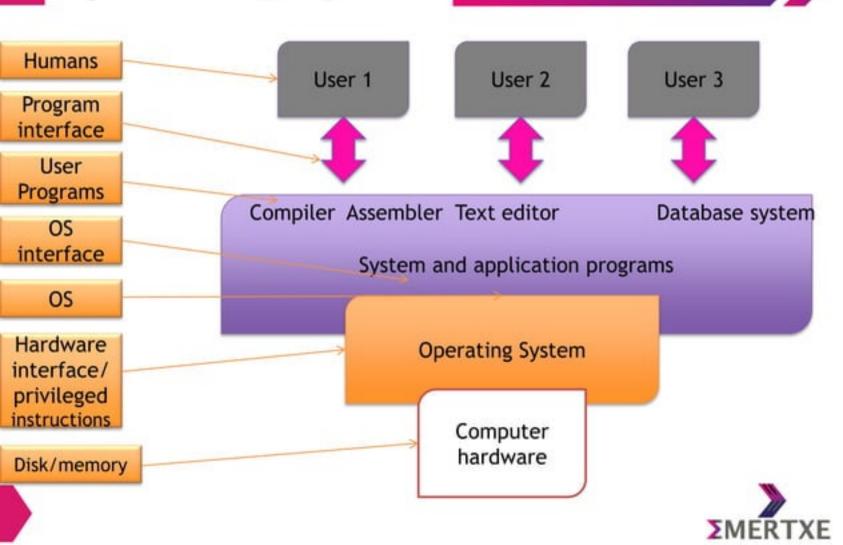
Linux Introduction

Let us ponder...

- ✓ What exactly is an Operating System (OS)?
- ✓ Why do we need OS?
- ✓ How would the OS would look like?
- ✓ Is it possible for a team of us (in the room) to create an OS of our own?
- ✓ Is it necessary to have an OS running in a Embedded System?
- ✓ Will the OS ever stop at all?



Operating System



What is Linux?

- Linux is a free and open source operating system that is causing a revolution in the computer world.
- Originally created by Linus Torvalds with the assistance of developers called community
- ✓ This operating system in only a few short years is beginning to dominate markets worldwide.



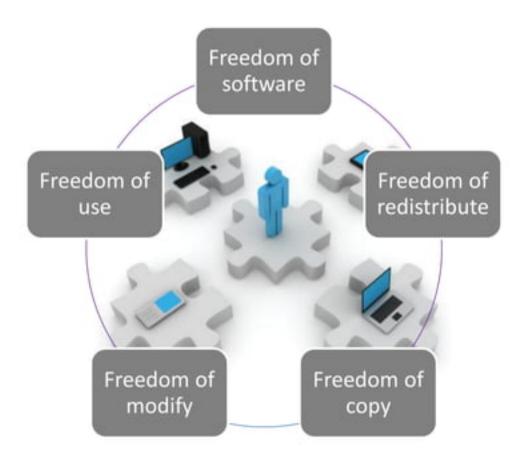


Why use Linux?

- ✓ Free & Open Source
- ✓ Reliability
- √ Secure
- √ Scalability



What is Open Source?





How it all started?

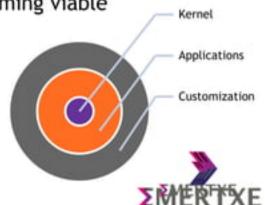
- ✓ With GNU (GNU is not UNIX)
- ✓ Richard Stallman made the initial announcement in 1983, Free Software Foundation (FSF) got formed during 1984
- ✓ Volunteer driven GNU started developing multiple projects, but making it as an operating system was always a challenge
- During 1991 a Finnish Engineer Linus Torvalds developed core OS functionality, called it as "Linux Kernel"
- ✓ Linux Kernel got licensed under GPL, which laid strong platform for the success of Open Source
- ✓ Rest is history!





How it evolved?

- ✓ Multiple Linux distributions started emerging around the Kernel
- ✓ Some applications became platform independent
- ✓ Community driven software development started picking up
- ✓ Initially seen as a "geek-phenomenon", eventually turned out to be an engineering marvel
- ✓ Centered around Internet
- ✓ Building a business around open source started becoming viable
- ✓ Redhat set the initial trend in the OS business



Where it stands now?





CIOSCUD

Novell

Databases







Server/Cloud









Enterprise









Consumer









Education







CMS



AUTOMATTIC



eCommerce





opencart ...



More details

Open Source SW vs. Freeware

OSS	Freeware
 ✓ Users have the right to access & modify the source codes ✓ In case original programmer disappeared, users & developer group of the S/W usually keep its support to the S/W. ✓ OSS usually has the strong users & developers group that manage and maintain the project 	 ✓ Freeware is usually distributed in a form of binary at 'Free of Charge', but does not open source codes itself. ✓ Developer of freeware could abandon development at any time and then final version will be the last version of the freeware. No enhancements will be made by others. ✓ Possibility of changing its licensing policy



GPL

- ✓ Basic rights under the GPL access to source code, right to make derivative works
- √ Reciprocity/Copy-left
- Purpose is to increase amount of publicly available software and ensure compatibility
- Licensees have right to modify, use or distribute software, and to access the source code



Problems with the GPL

- ✓ Linking to GPL programs
- ✓ No explicit patent grant
- ✓ Does no discuss trademark rights
- ✓ Does not discuss duration
- ✓ Silent on sub-licensing
- ✓ Relies exclusively on license law, not contract



Properties

- ✓ Multitasking
- ✓ Multi-user
- ✓ Multiprocessing
- ✓ Protected Memory
- √ Hierarchical File System



Components

- ✓ Hardware Controllers: This subsystem is comprised of all the possible physical devices in a Linux installation
- ✓ Linux Kernel: The kernel abstracts and mediates access to the hardware resources, including the CPU. A kernel is the core of the operating system
- ✓ O/S Services: These are services that are typically considered part of the operating system (e.g. shell)
- ✓ User Applications: The set of applications in use on a
 particular Linux system. (e.g. web-browser)



Directory structure

/boot/ Static files for boot loader /dev/ Device files /bin/ Essential user command binaries /etc/ Host specific system configuration /home/ User home directories /lib/ Essentilal shared libraries and kernal modules /media/ Mount point for removable media /mnt/ Mount point for temporarily mounted file systems /opt/ Add-on application software package /sbin/ User specific system binaries /tmp/ Temporary files /usr/ Multi-user utilities and applications /var/ Variable file (logs) /root/ Home directory for root user /proc/ Virtual file system documenting kernel and process status

Command Line Interface

Command Line Interface

- CLI
 - Textual mode
 - Executes requested command
- GUI
 - Mouse, keypad

```
| And the state of the state of
```





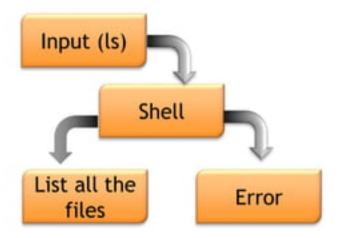
The Shell

- ✓ What is a shell?
- ✓ Different types of shells
 - Login-shell
 - · Non-login shell
 - Sh
 - Bash
 - Ksh
 - Csh
- ✓ Hands-on:
 - echo \$0
 - cat /etc/shells



How Shell Invokes

The main task of a shell is providing a user environment





Bash Files

✓ Bash

- Command interpreter
- .bash_profile (During login)
- .bashrc (New instance)
- .bash_logout (Logout)
- .bash_history (Command history)

√ Hands-on:

- Enter ls -a in your home directory
- Display contents of all files mentioned above



Environment Variables

- ✓ Login-shell's responsibility is to set the non-login shell and it will set the environment variables
- Environment variables are set for every shell and generally at login time
- ✓ Environmental variables are set by the system.
- ✓ Environmental variables hold special values. For instance ,\$ echo \$SHELL
- ✓ Environmental variables are defined in /etc/profile, /etc/profile.d/ and ~/.bash_profile.
- ✓ When a login shell exits, bash reads ~/.bash_logout



The 'bash' variables & Friends

- √ env lists shell environment variable/value pairs
- √ export [var_name] exports/sets a shell variable
 - HOME path to user's home directory
 - PATH executable search path
 - PWD present working directory
 - PS1 command prompt
- √ N=10 Assigning the variable. This a temporary variable effective only inside the current shell)
- ✓ unset N Unset the environment variable N



Basic Shell Commands

Basic Shell Commands

- √ \$ ls list's all the files
- √ \$ pwd gives present working directory
- √ \$ cd change directory
- √ \$ man gives information about command
- √ \$ exit exits from the shell



Shell: Built-in Commands

- ✓ Built-in commands are contained with in the shell itself, means shell executes the command directly, without creating a new process
- ✓ Built-in commands: break,cd,exit,pwd,export,return,unset,alias,echo,print f,read,logout,help,man



User Specific Command Set

- ✓ All Accesses into a Linux System are through a User
- ✓ User related Shell Command Set
- \$ useradd create user
- \$ userdel delete user
- \$ su [username] start new shell as different user
- \$ finger user information lookup
- \$ passwd change or create user password
- \$ who, w to find out who is logged in
- \$ whoami who are you



Remote login and remote copy

✓ ssh is a program for logging into a remote machine and for executing commands on a remote machine.

```
$ ssh ( secured login )
ssh <u>username@ipaddress</u>
```

√ scp copies files between hosts on a network.

```
$ scp ( secured copy )
scp filename username@ipaddress:/path/
```



File System Related Commands



- \$ stat File and Inode information
- \$ mount Mounting filesystem
- \$ find, locate Search for files



File Related Shell Commands

- Every thing is viewed as a file in Linux. Even a Directory is a file.
- ✓ Basic Shell Command Set

```
$ pwd - print working directory.
```

\$ cd - change directory.

\$ ls - list directory/file contents

\$ df - disk free

\$ du - disk usage

\$ cp - copy

\$ mv - move, rename

\$ rm - remove



Cont...

- √ \$ mkdir make directory
- √ \$ rmdir remove directory
- √ \$ cat, less, head, tail used to view text files
- √ \$ touch create and update files
- √ \$ wc counts the number of lines in a file



File Detailed Listing

group

```
aayush@aayush-laptop:~/Documents/try$ ls -l
total 4
brw----- 1 root root
                             7, 0 2010-09-12 02:02 block file
crw----- 1 root
                           108, 0 2010-09-12 02:02 character file
                    root
drwxr-xr-x 2 aayush aayush
                             4096 2011-03-17 03:56 directory file
                               12 2011-03-21 21:03 link file -> regular file
lrwxrwxrwx 1 aayush aayush
                                0 2011-03-17 04:51 namedpipe file
prw-r--r-- 1 root
                   root
rw-r--r-- 1 aayush aayush
                                0 2011-03-17 04:36 regular file
                                0 2011-03-17 04:32 socket file
srwxr-xr-x 1 aayush aayush
                                          Created time
                            File
                 Owner
                                                                 Filename
 permissions
                                            & Date
                            size
                   a
```



Linux file types

1st column

- -
- d
- (
- b
- |
- S
- = or p

Meaning

- Plain text
- Directory
- · Character driver
- Block driver
- Link file
- Socket file
- · FIFO file



File permissions

```
√r or 4 -r--r--
                        Read
√ w or 2 --w--w--
                        Write
√ x or 1 ---x--x
                        Execute
     rwx rwx rwx
     421
          421
              421
     user group others
Changing the File Permissions
$ chmod - Change file permessions
$ chown - Change file owner
$ chmod [ ug+r, 746 ] file.txt
$ chown -R user:group [ filename | dir ]
```



Redirection

- ✓ Out put redirection (>)
- √ Redirecting to append (>>)
- √ Redirecting the error (2>)

eg: \$ls > /tmp/outputfile

eg: \$ls -l >> /tmp/outputfile

eg: \$ls 2> /tmp/outputfile



Piping

- ✓ A pipe is a form of redirection that is used in Linux operating systems to send the output of one program to another program for further processing.
- A pipe is designated in commands by the vertical bar character

eg: \$ ls -al /bin | less



Other useful Command Set

Useful Command Set

- √ \$ uname print system information
- √ \$ man <topic> manual pages on <topic>
- √ \$ info <topic> information pages on <topic>
- √ \$ stat File and Inode information
- √ \$ find, locate Search for files
- √ \$ gzip filename This will compress folder or file
- √ \$ gunzip This will uncompress
- √ \$ tar Archiving files



Filters

- ✓ Filters are the programs, which read some input, perform the transformation on it and gives the output. Some commonly used filters are as follow
 - \$ tail : Print the last 10 lines of each FILE to standard output.
 - \$ sort : Sort lines of text files
 - \$ tr : Translate, squeeze, and/or delete characters from standard input, writing to standard output.
 - \$ wc : Print newline, word, and byte counts for each file



Pattern Matching

- ✓ Grep is pattern matching tool used to search the name input file. Basically its used for lines matching a pattern
 - Command: grep

Example: \$ ls | grep *.c

This will list the files from the current directory with .c extension



VIsual editor

VIsual editor

- √ vi or vim
- ✓ To open a file \$ vi <filename> or vim <filename>



VIsual editor...

- √ vi opens a file in command mode to start mode.
- ✓ The power of vi comes from its 3 modes
 - Escape mode (Command mode)
 Search mode
 File mode
 - Editing mode
 Insert mode
 Append mode
 Open mode
 Replace mode
 - Visual mode.



Cursor Movement

- ✓ You will clearly need to move the cursor around your file. You can move the cursor in command mode.
- ✓ vi has many different cursor movement commands. The four basic keys appear below
 - k move up one line
 - h line move one character to the left
 - I line move one character to the right
 - j move down one line
- ✓ Yes! Arrow keys also do work. But these makes typing faster

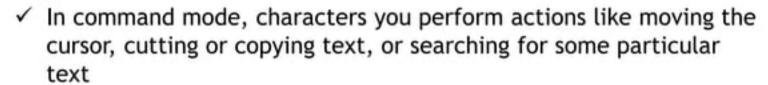


Basic vi commands

- ✓ How to exit
 - :q -> Close with out saving.
 - :wq -> Close the file with saving.
 - :q! -> Close the file forcefully with out saving
- ✓ Already looks too complicated?
- ✓ Try by yourself, let us write a C program
- ✓ Try out vimtutor. Go to shell and type vimtutor.



Escape mode or Command mode



Search mode

- vi can search the entire file for a given string of text. A string is a sequence of characters. vi searches forward with the slash (/) key and string to search. To cancel the search, press ESC . You can search again by typing n (forward) or N (backward). Also, when vi reaches the end of the text, it continues searching from the beginning. This feature is called wrap scan
- Instead of (/), you may also use question (?). That would have direction reversed
- Now, try out. Start vi as usual and try a simple search. Type
 /<string> and press n and N a few times to see where the
 cursor goes.

Escape mode...

√ File mode

- Changing (Replacing) Text
 - :%s/first/sec Replaces the first by second every where in the file
 - :%s/fff/rrrr/gc For all lines in a file, find string "fff" and replace with string "rrrrr" for each instance on a line
 - :q Close with out saving
 - :wq Close the file with saving
 - :q! Close the file forcefully with out saving
 - :e filename open another file without closing the current
 - :set all display all settings of your session
 - :r filename reads file named filename in place



Editing Modes...

✓ Command	Mode Name	Insertion Point
a	Append	just after the current character
Α	Append	end of the current line
i	Insert	just before the current character
-1	Insert	beginning of the current line
О	Open	new line below the current line
0	Open	new line above the current line



Editing Text

✓ Deleting Text Sometimes you will want to delete some of the text you are editing. To do so, first move the cursor so that it covers the first character of the group you want to delete, then type the desired command from the table below.

dd

For deleting a line

ndd

For deleting a n lines

X

- To delete a single character

shift + d

Delete contents of line after cursor

dw

Delete word's

ndw

Delete n words



Some Useful Shortcuts

- shift-g Go to last line in file
- shift-j Joining the two lines
- It repeats the previous command executed
- ctrl+a Increment number under the cursor
- ctrl+x Decrements numbers under the cursor



Visual Mode

√ Visual Mode

Visual mode helps to visually select some text, may be seen as a sub mode of the command mode to switch from the command mode to the visual mode type one of

- ctrl+v :- Go's to visual block mode.
- Only v for visual mode
- d or y Delete or Yank selected text
- I or A Insert or Append text in all lines (visual block only)



Stay connected

About us: Emertxe is India's one of the top IT finishing schools & self learning kits provider. Our primary focus is on Embedded with diversification focus on Java, Oracle and Android areas

Emertxe Information Technologies,
No-1, 9th Cross, 5th Main,
Jayamahal Extension,
Bangalore, Karnataka 560046
T: +91 80 6562 9666
E: training@emertxe.com







https://twitter.com/EmertxeTweet



https://www.slideshare.net/EmertxeSlides





Thank You