

COMP4137/COMP7200

Blockchain technology

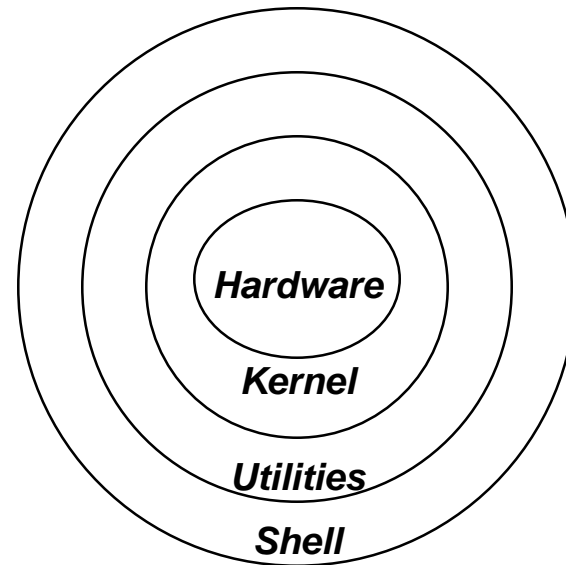
Tutorial 01: Linux – Basics



Structure of Linux

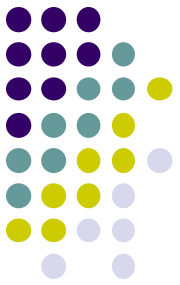


- Hardware
- Kernel
- Utilities
- Shell



Open terminal

- Display like this:
 - SEED@:~\$





Linux Commands

- `[SEED@~] $ command [-options] parameter1
parameter2`
- command: name of executed programs
- options: optional
- parameters: optional
- Note: most of commands are case sensitive



Basic Linux Commands

- clear – clears the screen
 - “*clear*” will give you a blank screen. You can still scroll up to see your text again in putty/ssh shell.
- cat – displays a file (all at once)
 - “*cat myfile*” will print out myfile to the screen.
- man – Displays a manual for any command
 - “*man pwd*” will give you the manual for the pwd command.
- exit – exits the terminal
 - “*exit*” will log you out and close the terminal.



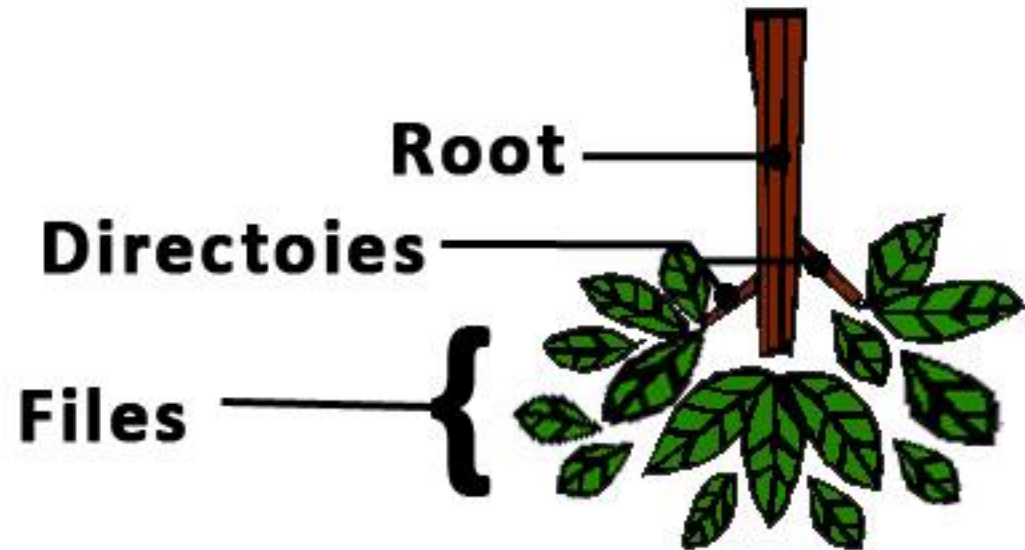
File System Structure

- To start out, you need to have an understanding of how Linux arranges it's files. So let's call the file system a tree.
- Think about hanging the tree upside down. All branches (directories) come from the trunk (root), or from other branches (directories).
- So if the branches are file folders (directories), then the leaves must be the files contained in the directories!



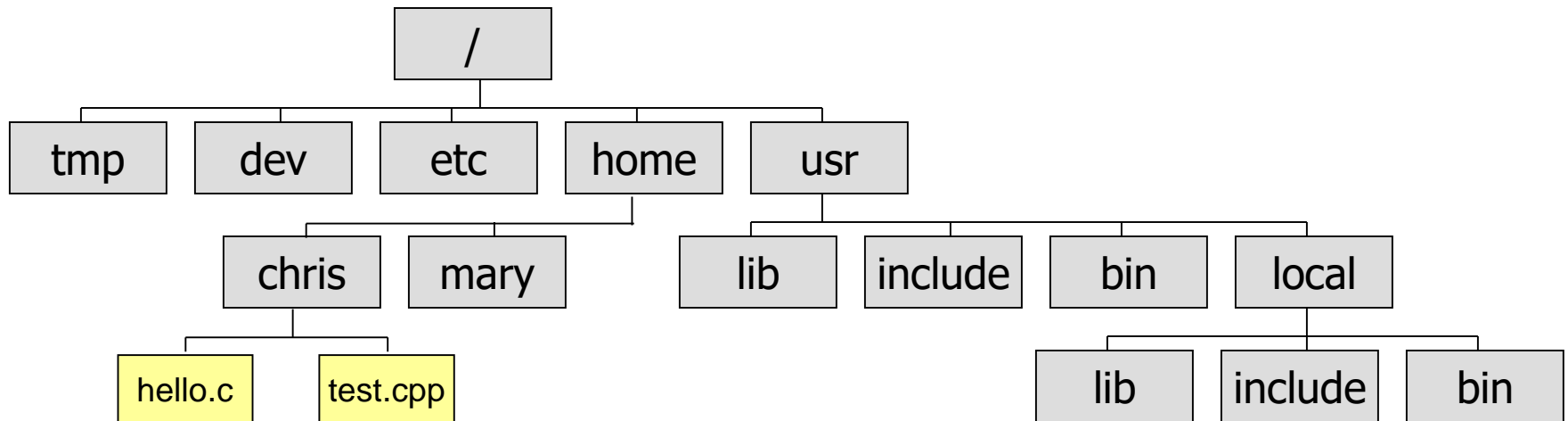
File System Structure

- The trunk is the root directory.
- Branches are directories.
- Leaves are files.





Example of UNIX file system



- Filenames case sensitive
- Physical devices can be mounted anywhere



File System Structure

- So does the file system structure of Unix look similar to another operating system?
- It should! It's very similar to any windows, mac, or linux operating system!
 - A path to a file from the root could look something like this:
`/home/seed/hello.c`
- So you should already know how things work. You just need to know the Unix commands to get around!

Note that Unix uses the forward slash! (Opposite of windows)



Handling Files and Directories

- **ls** : list files
- **cp** : copy files
- **mv** : move files
- **rm** : remove files
- **mkdir** : make directories
- **cd** : change directories
- **rmdir** : remove directories
- **pwd** : print working directory



ls command

- Syntax
 - ls [-Options] [name ...]
- Description
 - Lists contents of directory.
- Frequently Used Options
 - -a List all entries, including . and ..
 - -d Do not list contents of directories
 - -l Long listing
 - -F Mark directories with a '/', etc.
- Examples
 - ls -alF



cp command

- Syntax
 - `cp [-Options] file1 [file2 ...] target`
- Description
 - `File1` is copied to `target`.
- Frequently Used Options
 - `-f` Force remove existing file
 - `-i` Ask before removing existing file
 - `-r` Copy directory trees
- Examples
 - `cp p1.c p2.c`
 - `cp p1.c p2.c mydir`



mv command

- Syntax
 - `mv [-Options] file1 [file2 ...] target`
- Description
 - `File1` is moved to `target`.
- Frequently Used Options
 - `-f` Removes existing files without prompting the user
 - `-i` Asks before removing existing file
- Examples
 - `mv p*.c mydir`



rm command

- Syntax
 - `rm [-f] [-i] file . . .`
 - `rm -r [-f] [-i] dirname . . . [file . . .]`
- Description
 - Removes files or directories.
- Frequently Used Options
 - `-f` Removal of files without prompting the user
 - `-i` Interactive removal
 - `-r` Recursive removal
- Examples
 - `rm -f p*.o`
 - `rm -r mydir`



mkdir command

- Syntax
 - `mkdir [-m mode] [-p] dirname . . .`
- Description
 - Creates the specified directories.
- Options
 - `-m` Specifies the `mode` to be used
 - `-p` Create missing intermediate directories
- Examples
 - `mkdir -m 700 letter`
 - `mkdir abc`
 - `mkdir -p ./abc/def/ghi`



cd command

- Syntax
 - `cd [directory]`
- Description
 - Change working directory.
 - If directory is not specified, the value of shell parameter `$HOME` is used as the new working directory.
- Examples
 - `cd`
 - `cd ./abc/def/ghi`
 - `cd ..`



rmmdir command

- Syntax
 - `rmmdir [-p] [-s] dirname . . .`
- Description
 - Removes directories.
- Options
 - `-p` Remove the directory `dirname` and its parent directories which become empty.
 - `-s` Suppress the message when `-p` is in effect
- Examples
 - `rmmdir letter`



pwd command

- Syntax
 - pwd
- Description
 - Prints the path name of the working (current) directory.
- Examples
 - pwd



Directory

- Absolute directory
 - ▬ `/home/SEED/program`
- Relative directory: the directory relative to your current working directory
 - ▬ E.g., your current working directory: `/home/SEED/`
 - ▬ Then, you can use “`cd ../program`”, which is a relative directory.



Several directories:

- `.` current directory
- `..` parent directory
- `~` home directory of the current user.
 - E.g., `[SEED]$ cd ~/program`
 - Will change current working directory to `/home/SEED/program`



cat

- Display a file
- -E : end-of-line is displayed as \$;
- -n : display with line no. ;
- -T : [tab] is shown as ^I ;