LINUX

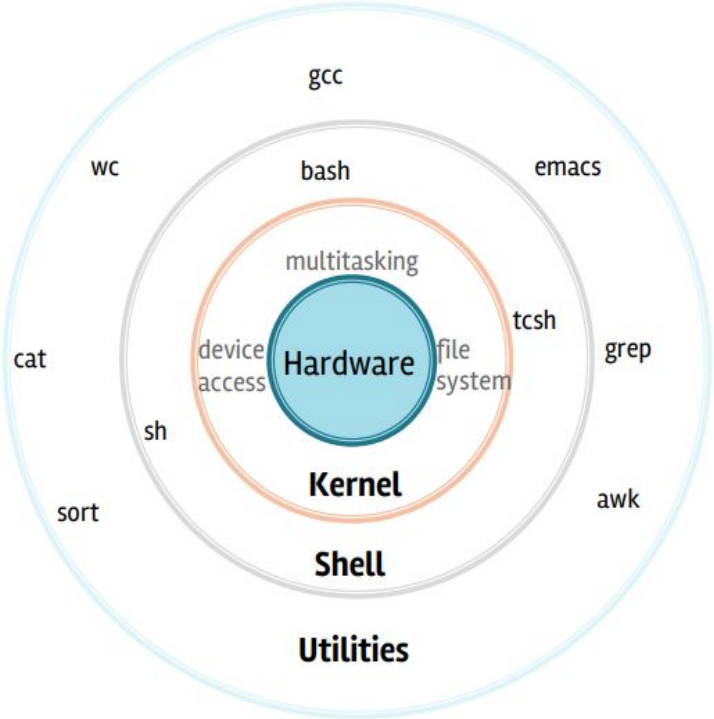
What is Linux ?

This language was especially developed for creating the UNIX system. Using this new technique, it was much easier to develop an operating system that could run on many different types of hardware.

The software vendors were quick to adapt, since they could sell ten times more software almost effortlessly. Weird new situations came in existence: imagine for instance computers from different vendors communicating in the same network, or users working on different systems without the need for extra education to use another computer. UNIX did a great deal to help users become compatible with different systems.

Comes in several “**distributions**” to serve different purposes.

LINUX ARCHITECTURE



Linux: The Shell

Program that interprets commands and sends them to the OS

- Provides:
  - Built-in commands
  - Programming control structures
  - Environment variables

- Linux supports multiple shells.
  - The default on SCC is Bash.

BASH => BOURNE AGAIN SHELL

Command Basics

- Command: Command/program that does one thing

- Options: Change the way a command does that one thing

    - Short form: Single-dash and one letter e.g. ls -a
    - Long form: Double-dash and a word e.g. ls --all

- Argument: Provides the input/output that the command interacts with.

COMMANDS

○ whoami	# my login
○ hostname	# name of this computer
○ echo “Hello, world”	# print characters to screen
○ echo $HOME	# print environment variable
○ echo my login is $(whoami)	# replace $(xx) with program output
○ date	# print current time/date
○ cal	# print this month’s calendar

LINUX COMMANDS BEGINNER MUST KNOW

pwd command

Use the **pwd** command to find out the path of the current working directory (folder) you're in.

cd command

To navigate through the Linux files and directories, use the **cd** command. It requires either the full path or the name of the directory, depending on the current working directory that you're in.

ls command

The **ls** command is used to view the contents of a directory. By default, this command will display the contents of your current working directory.

cat command

cat (short for concatenate) is one of the most frequently used commands in Linux. It is used to list the contents of a file on the standard output (sdout). To run this command, type **cat** followed by the file's name and its extension. For instance: **cat file.txt**.

cp command

Use the **cp** command to copy files from the current directory to a different directory. For instance, the command **cp scenery.jpg /home/username/Pictures** would create a copy of **scenery.jpg** (from your current directory) into the **Pictures** directory.

mv command

The primary use of the **mv** command is to move files, although it can also be used to rename files.

The arguments in mv are similar to the cp command. You need to type **mv**, the file's name, and the destination's directory. For example: **mv file.txt /home/username/Documents**.

mkdir command

Use **mkdir** command to make a new directory — if you type **mkdir Music** it will create a directory called **Music**.

rmdir command

If you need to delete a directory, use the **rmdir** command. However, rmdir only allows you to delete empty directories.

rm command

The **rm** command is used to delete directories and the contents within them. If you only want to delete the directory — as an alternative to rmdir — use **rm -r**.

touch command

The **touch** command allows you to create a blank new file through the Linux command line. As an example, enter touch **/home/username/Documents/Web.html** to create an HTML file entitled **Web** under the **Documents** directory.

grep command

Another basic Linux command that is undoubtedly helpful for everyday use is **grep**. It lets you search through all the text in a given file.

To illustrate, **grep blue notepad.txt** will search for the word blue in the notepad file. Lines that contain the searched word will be displayed fully.

chmod command

chmod is another Linux command, used to change the read, write, and execute permissions of files and directories. As this command is rather complicated, you can read **the full tutorial** in order to execute it properly.

history command

When you've been using Linux for a certain period of time, you'll quickly notice that you can run hundreds of commands every day. As such, running **history** command is particularly useful if you want to review the commands you've entered before.

echo command

This command is used to move some data into a file. For example, if you want to add the text, "Hello, my name is John" into a file called name.txt, you would type **echo Hello, my name is John >> name.txt**

THE END