Kernel vs. Operating System and Tools

Kernel

- The word **Linux** is often sloppily applied to the entire operating system and environment on computers which are equipped with a complete Linux distribution; but in fact, there are quite a few components which are necessary in order to have a fully functional platform

- Narrowly defined, Linux is only the **kernel** of the operating system (OS)
  - The kernel is the central component that connects the hardware to the software, and manages the system's resources, such as memory, CPU time sharing among competing applications and services
  - It handles all the devices that are connected to the computer by including so-called device drivers, and makes them available for the operating system to use

- A system running only a kernel has limited functionality, and the only place you will see that is in a dedicated device (often termed an **embedded device**) such as inside an **appliance**

Operating System

In order to do something useful and to be able to do a variety of things as needs arise, you need some other components, which in and of themselves are not strictly part of Linux:

- Important **system libraries**
  - Usually these are **shared libraries** or **dynamic linked libraries**
  - They can be used simultaneously by more than one program
  - The most important one is **libc**, which is used by virtually every application (among other things it handles the communication between the applications and the kernel)

- Important **system services** (sometimes called **daemons**)
  - Started when the system runs to control and monitor activities on the system, e.g. networking, printing, disk maintenance, noticing when new equipment is plugged in, monitoring system load and performance, etc.

Operating System (Cont.)

And,

- Basic **system utilities**
  - Those which handle listing files, viewing them, renaming them, removing them, etc.; bringing network connections up and down; compressing and decompressing files, etc.
  - Many of these utilities (which are often simple ones) are needed by the services already mentioned
  - A particularly important program is the command **shell** program, which is what users interact with when they work at a command line, but which is also used by non-interactive scripts
  - The default shell for Linux is usually **bash**, which stands for **B**ourne **A**gain **SH**ell, since it is an extension of the older **sh**, or Bourne Shell program

Main Components of Linux Operating System

Strictly speaking, the ingredients we discussed (kernel, including device drivers, services, and utilities) are enough to constitute a complete operating system, but other ingredients will be found on general service computers, as what we presented will only give you a **command-line based terminal**.
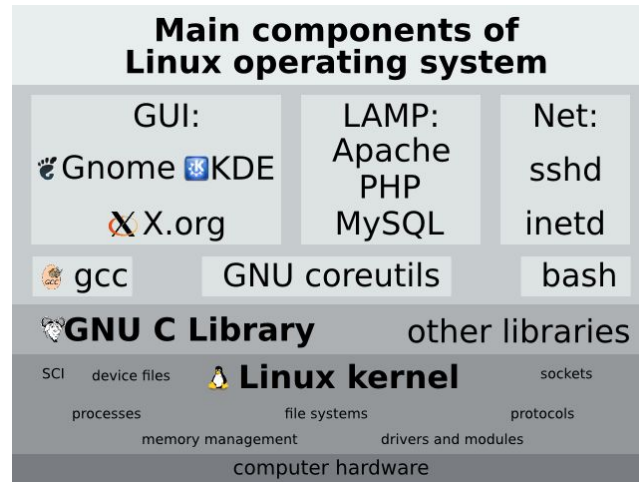
Graphical User Interface (GUI)

- Normal users will almost always be running a **Graphical User Interface** (GUI)

- Almost all desktop Linux systems will be built using the **X Window System** (or **X**) as the base of this interface; it has been around since at least 1984

- Besides **X**, there will be a so-called **window manager** which controls the appearances and behaviour of windows

- And a **desktop manager** which controls the entire desktop; the two most common choices in Linux are **GNOME** and **KDE**

Add-on Applications

- Many applications are part of the standard installation, but they are not part of the operating system - they are added on
  - The early versions of the Windows platform didn't contain many basic utility programs, and these had to be supplied by third-party providers until built-in implementations were deployed
  - The same is true of Internet browsers

- While many operating systems have now gotten so dependent on some of these programs being around that it is hard or impossible to operate without them, strictly speaking they are not absolutely required

- One advantage of Linux is that since there is so much choice for these components, they are used in a **modular way**, and the system can run with different or even no choices for them, unlike strictly controlled commercial platforms which either do not give such freedoms, or work hard to obscure and make it difficult to exercise options

Developmental Environment Tools

- Another important component is the **developmental environment tools**, such as compilers, debuggers, etc.

- While not all users need these, they are always available in Linux distributions

Linux Distribution

- It is the role of the **Linux distribution** to bring all these ingredients together in a coherent way

- Since they all do this somewhat differently, the look and feel can be quite dissimilar even when everything is built on top of the same Linux operating system kernel