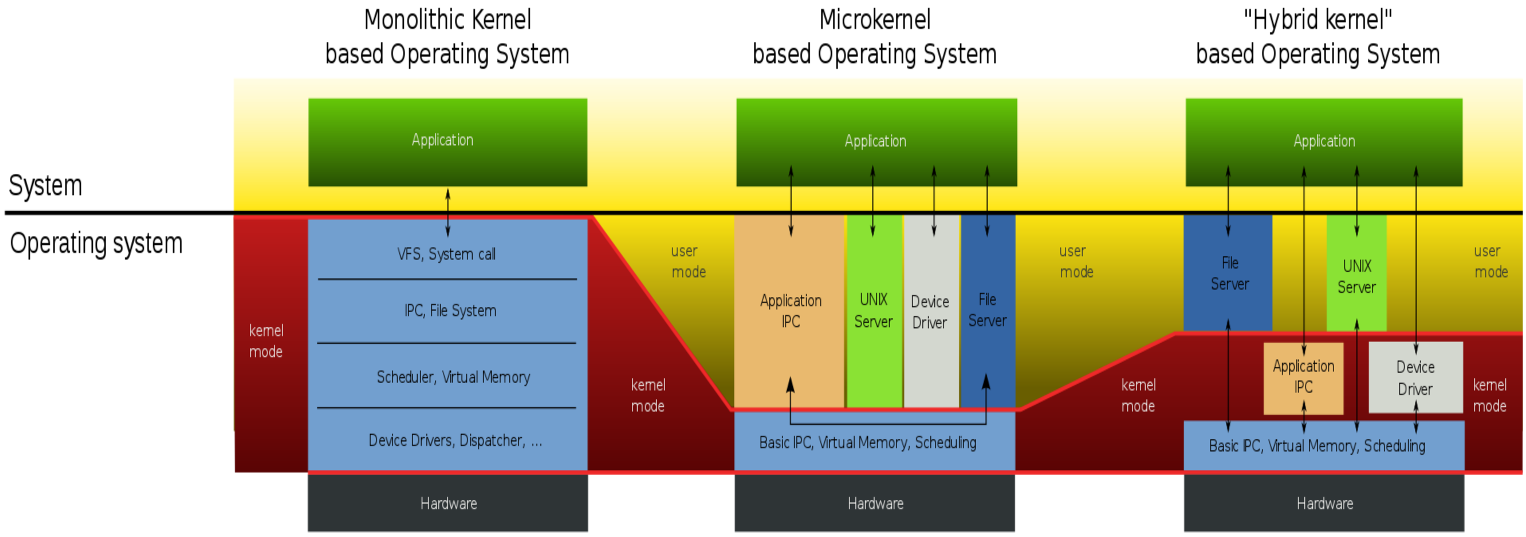
What is Kernel?

A Kernel is a computer program that is the heart and core of an Operating System. Since the Operating System has control over the system so, the Kernel also has control over everything in the system. It is the most important part of an Operating System. Whenever a system starts, the Kernel is the first program that is loaded after the bootloader because the Kernel has to handle the rest of the thing of the system for the Operating System. The Kernel remains in the memory until the Operating System is shut-down. The Kernel is responsible for low-level tasks such as disk management, memory management, task management, etc. It provides an interface between the user and the hardware components of the system. When a process makes a request to the Kernel, then it is called System Call. A Kernel is provided with a protected Kernel Space which is a separate area of memory and this area is not accessible by other application programs. So, the code of the Kernel is loaded into this protected Kernel Space.

DIFFERENT KERNEL ARCHITECTURES

1. Monolithic 2. Micro Kernel 3. Hybrid Kernel



Monolithic Kernel: Earlier in this type of kernel architecture, all the basic system services like process and memory management, interrupt handling etc were packaged into a single module in kernel space. This type of architecture led to some serious drawbacks like:

Size of kernel, which was huge.

Poor maintainability, which means bug fixing or addition of new features resulted in recompilation of the whole kernel which could consume hours.

In modern day approach to monolithic architecture, the kernel consists of different modules which can be dynamically loaded and un-loaded. This modular approach allows easy extension of OS's capabilities. With this approach, maintainability of kernel became very easy as only the concerned module needs to be loaded and unloaded every time there is a change or bug fix in a particular module.

So, there is no need to bring down and recompile the whole kernel for a smallest bit of change. Also, stripping of kernel for various platforms (say for embedded devices etc.) became very easy as we can easily unload the module that we do not want.

Linux follows the monolithic modular approach.

Microkernels: This architecture majorly caters to the problem of ever-growing size of kernel code which we could not control in the monolithic approach.

This architecture allows some basic services like device driver management, protocol stack, file system etc to run in user space. This reduces the kernel code size and also increases the security and stability of OS as we have the bare minimum code running in kernel. So, if suppose a basic service like network service crashes due to buffer overflow, then only the networking service's memory would be corrupted, leaving the rest of the system still functional.

In this architecture, all the basic OS services which are made part of user space are made to run as servers which are used by other programs in the system through inter process communication (IPC). For example: we have servers for device drivers, network protocol stacks, file systems, graphics, etc. Microkernel servers are essentially daemon programs like any others, except that the kernel grants some of them privileges to interact with parts of physical memory that are otherwise off limits to most programs. This allows some servers, particularly device drivers, to interact directly with hardware. These servers are started at the system start-up.

Hybrid kernel: The hybrid approach is derived from the best of both micro and monolithic kernel architectures. Instead of loading the whole thing into memory, core modules are loaded dynamically to memory on demand. One disadvantage is that a module may destabilize a running kernel.

KERNAL IN LINUX

1. Linux Kernel, which is in the Linux Operating System, is an open source software based on Unix.
2. Linux chooses to keep its configurations into files.
3. Linux has segmented working environments which secure it from the attack of virus.
4. Linux offers 100% multi-user environments.
5. Linux kernel has monolithic architecture.
6. The Linux kernel keeps the GUI stack in the user-space.

KERNAL IN WINDOWS

1. Windows kernel, which is in Windows Operating System, is a commercial software developed by Microsoft.

2. The Windows kernel keeps a registry to store its configurations.

3.Windows is not much segmented and thus it is more vulnerable to threats.

4.Windows kernel offer multi-user and multi-session support, but it depends on the version and edition of your Windows OS.

5.Windows kernel has hybrid architecture.

6.The Windows kernel includes a GUI stack in the kernel.