Introduction

(Kernel definition, functions, types)

Good afternoon everyone. I am shreyansh, he is shamim and he is shivanshu. We are here to give a presentation on COMPILATION OF KERNEL. But before we start, we must know what is kernel and why do we need to compile it?

Kernel simply means “the central or most important part of something.” But according to computer science, it means “a interface or layer between hardware and operating system.” It is the most fundamental part of operating system. It can be thought of as the program which controls all other program on the computer.

Typically it is responsible for memory management, process and task management and disk management. It makes these facilities available to application processes through inter process communication mechanisms and system call.

It takes responsibility for deciding at any time which of the many running programs should be allocated to the processor.

It decides which memory each process can use. Memory is used to store both program instructions and data. Multiple programs want access to memory, frequently demanding more memory than the computer has. It determines what to do when enough memory is not available.

It allocates requests from applications to perform I/O to an appropriate device.

Operating system tasks are done differently by different kernel depending on their design and implementation.

Monolithic kernels execute all the operating system code in same address space to increase the performance of the system. All OS services run along with the main kernel thread thus residing in the same memory area. This approach provides rich and powerful hardware access.

Micro kernels run most of the operating system services in user space as server aiming to improve maintainability and modularity of the operating system. They are easier to maintain than monolithic kernels but the large number of system calls and context switching might slow down the system because they generate more overhead than plain function calls.

Kernel compilation

I don’t think that anyone of you has ever compiled a kernel before this lab. So the first reason can be, this lab demands it.

Second reason can be that you want to upgrade your kernel and try some new features

Third reason can be that you have installed a new hardware and your existing kernel doesn’t support it that’s why you need new kernel.

Fourth reason can be that you want to debug the kernel.

And the fifth and most interesting reason can be that you are a adventurous person and want to simply learn how to compile a kernel.

Whatever the reason may be, we will demonstrate the compilation of kernel.