**Tutorial 13**

Una vez dado por ejemplificado los conceptos mas fundamentales de un kernel, es necesario ahondar en cada uno de ellos. El kernel primeramente debe ofrecer protección de memoria y manejo de memoria. Son dos temas muy diferentes. Por ejemplo, la protección de memoria debe: proteger el sistema en caso de que un programa acceda a un descriptor equivocado, proteger que el programa no se sobrescriba o que el programa acceda a la memoria de otro programa. El manejo de memoria debe: asignar memoria dinámicamente para que el programa se ejecute, implementar el pagina y memoria virtual para los programas, asegurarse que el kernel no escriba en una dirección de memoria desconocida, manejar el fragmentado de memoria.

Existe también el manejo de procesadores multinucleo o multiprocesadores. La BIOS se encarga de levantar el primero núcleo de un procesador de este tipo, pero es nuestro sistema operativo el que se debe encargar de levantar los demás a funcionar y manejar los procesos y como se ejecutan dentro de ellos. Otra cosa es el manejo de los programas en si, como ejecutar programas simultáneamente, proteger que los programas no accedan a archivos o memoria que no deben, asignar y quitar memoria dinámicamente a los programas, etc.

Las llamadas al sistema deben de ser propiamente implementadas por los sistemas operativos para dar acceso a los programas de usuario a los recursos de hardware. Para esta forma, es mejor que los sistemas operativos provean una librería de programación y hacerle la vida mas fácil a los usuarios proveyendo esa capa de abstracción entre sistema operativo y programa mediante una liberaría estándar propia del sistema.

Existen diferentes tipos de kernels y cada uno empaqueta sus funciones de forma diferente. En primera instancia esta un kernel del tipo monolítico y este es un kernel que empaqueta todo su código, funciones y manejo de hardware dentro de un mismo programa. El problema con este kernel es que si logra fallar un tan solo servicio, todo el sistema se viene abajo. Este tipo de kernel no es recomendado y no es utilizado ya por los problemas que pueden ocasionarse, sin embargo, bien implementado daría mucha mas velocidad que otro tipo de kernel ya que todo esta empaquetado en un solo programa.

Otro tipo de kernel es el microkernel y este se diferencia bastante del otro tipo de kernel porque en vez de ser un solo kernel empaquetado, este separa sus servicios en servicios que corren a nivel de usuario. Por ejemplo, los drivers de hardware se ejecutan a diferentes niveles que el código de kernel. Si ocurre un tan solo fallo en un driver, el sistema puede manejar el error y recuperarse de algún forma. Esto ofrece estabilidad y modularidad al sistema operativo.

Por ultimo, existe otros dos tipos de kernels que son los kernels híbridos y los nanokernels. El kernel hibrido es una mezcla entre microkernel y kernels monoliticos, ciertos servicios están empaquetados mientras otros corren por separado. Los nanokernels simplemente establecen muy pocos servicios pero esenciales para que un sistema operativo pueda funcionar. Sin embargo, debido a que son muy pequeños, puede ser necesario tercerizar servicios importantes a otros programas.