El Kernel

El **kernel** es **la capa de software de más bajo nivel** en la computadora.
Si se define al sistema operativo como *el software que maneja y dispone de los recursos de una computadora*, entonce el término **kernel** puede ser equivalente al de **sistema operativo**.
Viendo al sistema como un conjunto de capas, el **sistema operativo** se denomina comúnmente **kernel del sistema**, o simplemente **kernel**, lo que enfatiza su aislamiento de los programas de los usuarios.

Tareas específicas del Kernel:

- **Planificar la ejecución de las aplicaciones**.
- **Gestionar la Memoria**.
- **Proveer un sistema de archivos**.
- **Creación y finalización de procesos**.
- **Acceder a los dispositivos.**
- **Comunicaciones.**
- **Proveer un API**.

Ejecución Directa

La **ejecución directa** de un programa significa, **correr el programa directamente en la CPU**

OS	Program
Create entry for process list	
Allocate memory for program	
Load program into memory	
Set up stack with argc/argv	
Clear registers	
Execute **call** main()	
	Run main()
	Execute **return** from main
Free memory of process	
Remove from process list	

Figure 6.1: **Direct Execution Protocol (Without Limits)**

La ventaja que tiene la ejecución directa es la rapidez. Pero tiene problemas:

1. Cuando se corre un programa, como se asegura el Sistema Operativo, que el programa no va a hacer nada que el usuario no quiere que haga?
2. Como hace el sistema Operativo para pausar la ejecución de ese programa y hacer que otro sea ejecutado?

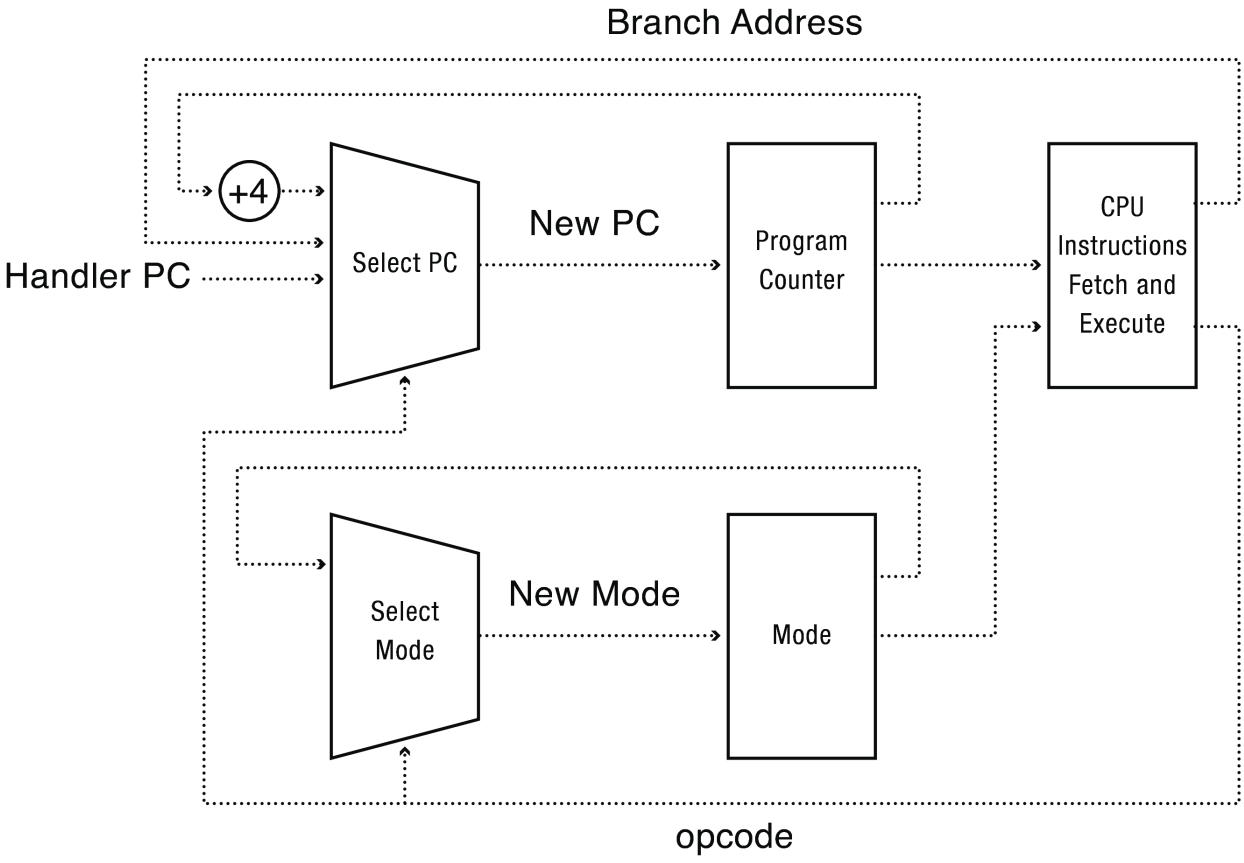
Limitar la Ejecución Directa

Por los problemas mencionados se necesita **Limitar la Ejecución Directa**. Para ello se necesitan ciertos **mecanismos de hardware**:

- **Dual Mode Operation** - Modo de operación dual.
- **Privileged Instructions** - Instrucciones Privilegiadas.
- **Memory Protection** - Proteccion de Memoria.
- **Timer Interrupts** - Interrupciones por temporizador.

Modo dual de operaciones

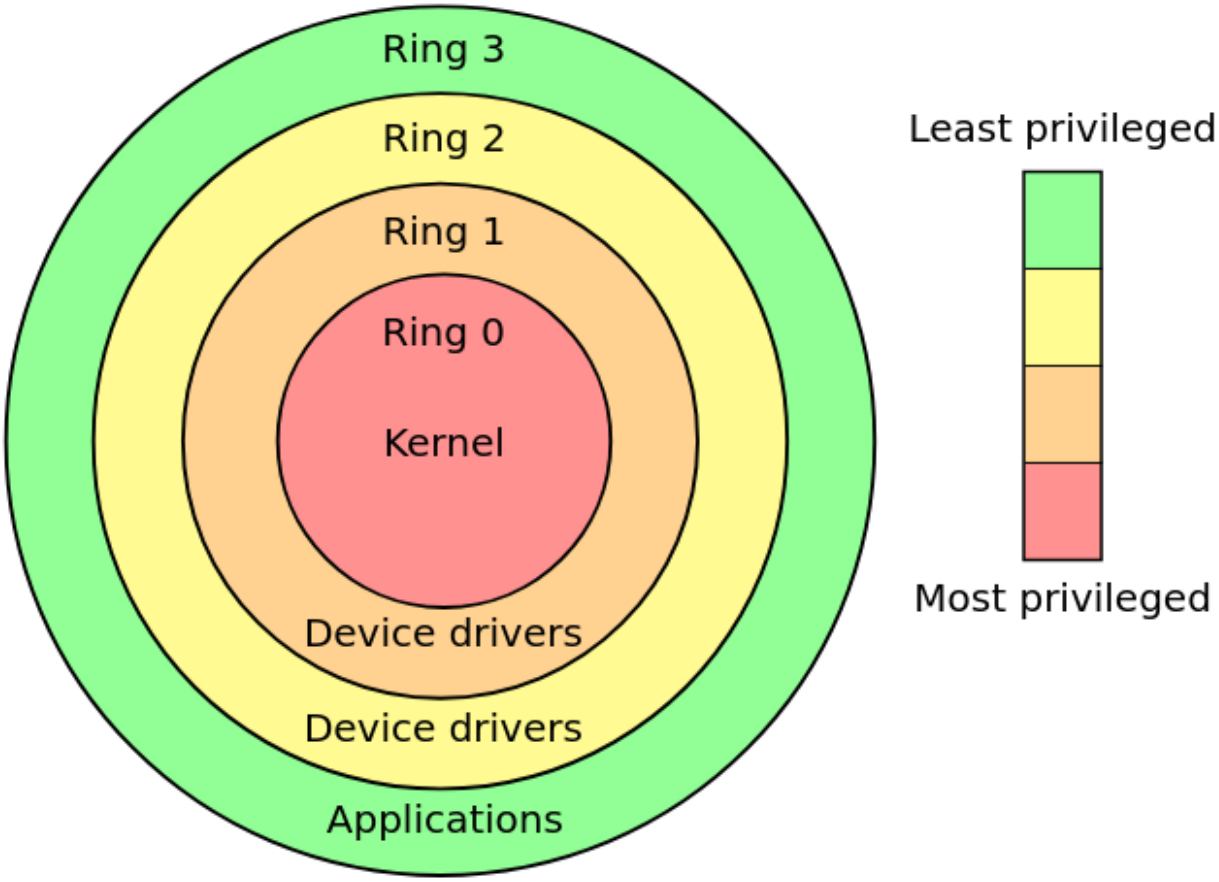
El funcionamiento en modo dual está diseñado para proporcionar una capa de protección y estabilidad a los sistemas informáticos separando los programas de usuario y el sistema operativo en dos modos: **modo usuario** y **modo kernel**.



Kernerl Land y User Land

Los modos

En el caso de la arquitectura x86 la misma provee 4 modos de operaciones vía el hardware. Estos modos están numerados entre 0 y 3, y se denominan **rings**.



Existen dos modos operacionales utilizados de la CPU :

- **Modo Usuario o User Mode** (modo 3): que ejecuta instrucciones en nombre del usuario.
- **Modo Supervisor o Kernel** (modo 0): ejecuta instrucciones en nombre del Kernel del S.O. y estas son instrucciones privilegiadas.

Se protege:

1. La memoria
2. Los Puertos de I/O
3. La Posibilidad de ejecutar ciertas instrucciones

Instrucciones privilegiadas

La existencia del mecanismo llamado **modo dual** permite que los distintos modos posean cada uno su **propio set de instrucciones**. Con lo cual el bit de modo de

operación indica al procesador si la instrucción puede ser o no ejecutada, según el modo en que se encuentre el mismo.

Protección de memoria

El sistema operativo y los programas que están siendo ejecutados por el mismo deben residir **ambos en memoria al mismo tiempo**.
El sistema operativo tiene que estar ahí para cargar el programa y hacer que comience a ejecutarse.
El programa tiene que residir en memoria para poder ejecutarse, es más todos los programas que se están ejecutando deben estar cargado en la memoria de la máquina.
Debido a esto, para que la memoria sea compartida de forma segura, el sistema operativo debe poder configurar al hardware de forma tal que cada proceso pueda leer y escribir su propia porción de memoria.

Timer interrupts

Para que el kernel pueda tomar el control de la computadora debe haber algún mecanismo que periódicamente le permita al kernel **desalojar** al proceso de usuario en ejecución y volver a tomar el control del procesador, y así de toda la máquina.
En la actualidad, todos los procesadores poseen un mecanismo de hardware llamado **hardware counter**, el cual puede ser seteado para que luego del transcurso de un determinado tiempo el procesador sea interrumpido. Cada CPU posee su propio timer.
Cuando una interrupción por tiempo ocurre, el proceso en modo usuario que se esté ejecutando le transfiere el control al kernel ejecutándose en modo kernel. De esta forma el kernel tiene asegurado el uso del procesador.

Modos de Transferencia

Una vez que el hardware posee los mecanismos necesarios para que pueda ejecutarse un kernel tiene que haber una o varias formas de alternar entre **modo usuario** y **modo kernel**.
Deben tener un mecanismo que sea seguro y rápido y que además no de lugar para programas maliciosos o con errores que pueden intencionalmente ser inseridos y corromper el Kernel.

De Modo Usuario a Modo Kernel

Existen tres formas por las cuales se debería pasar de pasar de modo usuario a modo kernel:

- interrupciones (evento externo),

- excepciones del procesador (evento interno),
- y mediante la ejecución de system calls (evento intencional).

Interrupciones

Una **interrupción** es una **señal asincrónica** enviada hacia el procesador de que algún **evento externo ha sucedido** y pueda **requerir de la atención** del mismo.

El procesador está continuamente chequeando por si una interrupción se dispara. Si así es, este **completa o detiene** cualquier instrucción que se esté ejecutando y en vez de ejecutar la siguiente instrucción, el procesador **guarda** todo el contexto en el que se estaba ejecutando la instrucción y comienza a ejecutar el manejador de esa interrupción en el kernel.

Algunas interrupciones son:

- Errores de la Máquina
- Timers
- Discos
- Network devices
- Terminales
- Interrupciones de Software

Excepciones del Procesador

La otra forma por la cual se necesitaría pasar de modo usuario a modo kernel es por un **evento de hardware causado por un programa de usuario**. El funcionamiento es igual de la una interrupción. Una excepción podría ser:

- Acceder fuera de la memoria del proceso
- Intentar ejecutar una instrucción privilegiada en modo usuario.
- Intentar escribir en memoria de solo lectura.
- dividir por cero.

System Calls

Las System Calls son funciones que permiten a los procesos de usuario pedirle al kernel que **realice operaciones** en su nombre. Una System Call es cualquier función que el el kernel **expone** que puede ser utilizada por un proceso a nivel usuario.

Una **system call** (llamada al sistema) es un punto de entrada controlado al kernel, permitiendo a un proceso solicitar que el **kernel** realice alguna operación en su

nombre. El kernel expone una gran cantidad de servicios accesibles por un programa vía el **application programming interface (API) de system calls**.

Algunas características generales de las system calls son:

- Una system call cambia el modo del procesador de *user mode* a *kernel mode*, por ende la CPU podrá acceder al área protegida del kernel.
- El conjunto de system calls es fijo. Cada system call esta identificada por un único número, que por supuesto no es visible al programa, este sólo conoce su nombre.
- Cada system call debe tener un conjunto de parámetros que especifican información que debe ser transferida desde el *user space* al *kernel space*.

De Modo Kernel a Modo Usuario

También hay varias formas de pasar de **modo kernel** a **modo usuario**:

- Un **nuevo proceso**. Cuando se inicia un nuevo proceso, el Kernel copia el programa en la memoria, setea el contador de programa apuntando a la primera instrucción del proceso, setea el stack pointer a la base del stack de usuario y switchea a modo usuario.
- **Continuar despues de una interrupcion, una excepcion del procesador o una system call**. Una vez que el Kernel termino de manejar el pedido, este continua con la ejecución de proceso interrumpido mediante la restauración de todos los registros y cambiando el modo a nivel usuario.
- **Cambio entre diferentes procesos**. En algunos casos puede pasar que el Kernel decida ejecutar otro proceso que no sea el que se estaba ejecutando, en este caso el Kernel carga el estado del proceso nuevo a través de la PCB y cambia a modo usuario.