

# Informe de proso

En esta practica el objetivo es realizar una modificación al sistema añadiendole la capacidad de guardar estadísticas.

Estas estadísticas que queremos conocer son para las llamadas a sistema open, write, clone, close y lseek.

Queremos saber el tiempo media de respuesta del sistema centrandonos en las llamadas anteriores, y para ellos contabilizaremos:

- las veces que se inicia la llamada
- veces en las que finaliza correctamente
- veces en las que finaliza incorrectamente
- tiempo total de ejecución de la llamada

Queremos realizarlo de forma dinámica, sin necesidad de recompilar el kernel, para poder activar o desactivar estas funcionalidades.

Para ello, emplearemos módulos, en concreto 2. el módulo 1 se encarga de interceptar las llamadas y contabilizar el tiempo y el módulo 2 se encarga de acceder a las estadísticas.

Para el módulo 1, hemos de substituir las funciones originales en la tabla de llamadas a sistema por unas locales y guardar las originales en un vector al iniciar el módulo por ejemplo:

```
sys_open_old [5];  
sys_open_old[x] = sys_call_table [posicion de la llamada en el sys call table];  
sys_call_table [posicion de la llamada en el sys call table] =  
sys_call_local_especifico;
```

y deshacerlo al eliminar el módulo.

las funciones específicas seguirán el siguiente esquema:

- obtener el tiempo inicial (usando la función proso\_get\_cycles)
- ejecutar la llamada original
- calcular tiempo de ejecución (restar tiempo actual a tiempo inicial)
- analizar el código de retorno de la llamada original
- guardar datos estadísticos
- retornar código de retorno de la llamada original.

Para almacenar las estadísticas, hemos de crear una estructura llamada "my\_thread\_info" que contenga la antigua estructura thread\_info y las estadísticas a continuación, y en estas estadísticas, hemos de añadir el pid del proceso asociado a las estadísticas para comprobar que no son datos reutilizados.

Al desinstalar el módulo, hemos de imprimir por pantalla las estadísticas del proceso con pid que hemos pasado como argumento al instalar el módulo.

También hemos de evitar el poder eliminar el módulo 1 si hay algún proceso con alguna llamada interceptada por este.

En cuanto al módulo 2, se dedicará a consultar la información guardada previamente. Para ello hemos de crear un nuevo dispositivo con su mayor y menor diferentes del resto y que permiten identificarlo y que utilizaremos para registrar el driver y el archivo que lo represente.  
podremos elegir el proceso del cual queremos información