

Sistemas Operativos

Práctica 6: Sistemas de archivos y protección de la información

Notas preliminares

- Los ejercicios marcados con el símbolo ★ constituyen un subconjunto mínimo de ejercitación. Sin embargo, aconsejamos fuertemente hacer todos los ejercicios.

Parte 1 – Sistemas de archivos

Ejercicio 1 ★

Suponer una computadora cuyo disco se accede sin memoria *cache* y un sistema de archivos FAT. Además, en este sistema, la FAT no queda almacenada en la memoria (recordar que lo normal es que la FAT se cargue en memoria). ¿Cuántos accesos al disco son necesarios para llegar hasta el último bloque de un archivo de N bloques?

Ejercicio 2 ★

Suponer que se cuenta con un sistema de archivos basado en inodos y bloques de 4 KB.

- Si se tiene un archivo de 40 KB, ¿cuántos bloques es necesario leer para procesarlo completamente?
- ¿Cuántos bloques es necesario leer si el archivo tiene 80 KB?

Ejercicio 3 ★

Una compañía que fabrica discos rígidos decide, como parte de cierta estrategia comercial, emprender la creación de un nuevo *filesystem*. Durante la fase de diseño preliminar los ingenieros a cargo del proyecto discuten acaloradamente la conveniencia de adoptar un enfoque inspirado en FAT o la de uno basado en inodos. Indicar cuál de las dos opciones recomendaría, y por qué, para cada uno de los siguientes requerimientos:

- Es importante que puedan crearse enlaces simbólicos.
- Es importante que la cantidad de sectores utilizados para guardar estructuras auxiliares sea acotada, independientemente del tamaño del disco.
- Es importante que el tamaño máximo de archivo sólo esté limitado por el tamaño del disco.
- Es importante que la cantidad de memoria principal ocupada por estructuras del *filesystem* en un instante dado sea (a lo sumo) lineal en la cantidad de archivos abiertos en ese momento.

Ejercicio 4 ★

Se tiene un disco rígido de 16 GB de espacio con sectores de 1 KB. Se desea dar formato al disco usando un sistema de archivos de uso específico llamado *HashFS*, basado en FAT. La idea es que no existen directorios ni archivos. Dado un path, se calcula el *hash* del nombre y éste indica cuál es el archivo buscado.

En resumen, este sistema de archivo cuenta con dos tablas:

- Una única FAT que guarda las entradas correspondientes al próximo bloque, indicando el final de un archivo cuando estos valores coinciden.
- Una única tabla de *hash* que contiene, para cada *hash* posible, el identificador del bloque inicial y el tamaño en bytes del archivo correspondiente a dicho *hash*.

La novedad es que este sistema de archivos permite configurar los siguientes elementos:

- Tamaño del bloque: 2, 4 u 8 sectores.
 - Tamaño de identificadores de bloque: 8, 16, 24 o 32 *bits*.
 - Tamaño del *hash*: 8, 16, 24 o 32 *bits*.
- a) Suponiendo que se configura con 2 sectores por bloque, identificadores de bloque de 24 *bits*, y *hash* de 16 *bits*. ¿Cuál es el tamaño que ocupa la FAT? ¿Cuál es el tamaño de la tabla de archivos? ¿Cuál es el espacio que queda en disco para archivos?
- b) Sabiendo que se desea maximizar la cantidad de archivos que el sistema soporta y que, además, en promedio los archivos tendrán un tamaño de 1 KB, ¿cuál sería la configuración óptima del sistema de archivos? Justificar.
- c) ¿Cómo lo configuraría si el promedio de tamaño de archivos es de 16 MB? Justificar.

Ejercicio 5

Linux permite pasar un descriptor de archivo de un proceso a otro (es decir, el valor de retorno de la syscall `open()`). Asuma que un proceso abre el directorio `/home` que está sobre una partición de Ext2 y le envía el descriptor del archivo `home` a otro proceso con los permisos necesarios para poder leerlo.

Asumiendo que tiene la función auxiliar `Ext2FSInode load_inode(int inode_number)` que dado un número de inodo devuelve la estructura del inodo, escriba el pseudocódigo que le permita obtener el nombre del directorio (`home`) a partir del descriptor de archivo recibido.

Ejercicio 6

Resolver para un sistema ext2 (12 entradas directas, 3 indirectas, 1 doble indirecta y 1 triple indirecta) y para un sistema basado en FAT:

¿Cuántos bloques de disco tengo que acceder para acceder a los siguientes bloques de un archivo? Asumir si un bloque se lee dos veces, se va a buscar una sola vez al disco y que tanto la FAT como el inodo del archivo correspondiente ya están cargados en memoria.

- a) 1, 2, 3, 5, 7, 11, 13, 17, 23.
- b) 1, 2, 3, 4, 5, 6, 10001.
- c) 13, 10000, 1000000.
- d) 14, 15...50.

Parte 2 – Redundancia de disco y resguardo de datos

Ejercicio 7 ★

¿Podría en algún contexto un esquema de organización de RAID nivel 1 proporcionar un mejor rendimiento para las solicitudes de lectura que un esquema RAID nivel 0?

Ejercicio 8 ★

Considerar un esquema de organización RAID nivel 5 compuesto por 5 discos; es decir, que cada 4 bloques almacenados en 4 discos, el disco restante almacena la información de paridad. ¿A cuántos bloques hay que acceder para llevar a cabo las siguientes operaciones?

- a) Una escritura de un bloque de datos.
- b) Una escritura de siete bloques contiguos de datos.

Ejercicio 9 ★

Una estrategia de copias de seguridad (“*backup*”) en cintas utiliza 5 cintas que va rotando. Responder y justificar cuántas cintas puede llegar a ser necesario utilizar para restaurar un archivo en cada uno de los siguientes casos:

- a) Total.
- b) Diferencial.
- c) Incremental.

Ejercicio 10 ★

Un sistema cuenta con almacenamiento basado en RAID 5. Responda verdadero o falso justificando:

- a) Realizar copias de seguridad a cinta es innecesario ya que no incrementa el nivel de protección.
- b) Desde el punto de vista de la seguridad informática, realizar copias de seguridad a cintas aumenta la exposición a riesgos de los datos.

Ejercicio 11

Explique qué es un *snapshot* de un *filesystem*, y cómo se relaciona con una estrategia de copias de seguridad.