

## PRACTICA 04

Código: 20120354I

Apellidos y Nombres: **Moreno Vera Felipe Adrian**

1. En los primeros sistemas UNIX, los archivos ejecutables (archivos *a.out*) empezaban con un número mágico muy específico, no uno elegido al azar. Estos archivos empezaban con un encabezado, seguido de los segmentos de texto y de datos. ¿Por qué cree usted que se eligió un número muy específico para los archivos ejecutables, mientras que otros tipos de archivos tenían un número mágico más o menos aleatorio como la primera palabra?

**Sol:**

Los sistemas cargados directamente al programa en memoria, inician ejecutando la palabra 0, el cual fue conocido como el “magic number”. Para evadir el intento de ejecutar la cabecera como código, el “magic number” fue una instrucción **BRANCH** con una dirección objetivo justo por encima de la cabecera. De esta forma fue posible leer el archivo binario directamente en la dirección del espacio del nuevo proceso y ejecutarlo en 0, sin siquiera saber qué tan grande fue la cabecera.

2. En la figura 4-4, uno de los atributos es la longitud del registro. ¿Por qué se preocupa el sistema operativo por esto?

Atributo	Significado
Protección	Quién puede acceso al archivo y en qué forma
Contraseña	Contraseña necesaria para acceder al archivo
Creador	ID de la persona que creó el archivo
Propietario	El propietario actual
Bandera de sólo lectura	0 para lectura/escritura; 1 para sólo lectura
Bandera oculto	0 para normal; 1 para que no aparezca en los listados
Bandera del sistema	0 para archivos normales; 1 para archivo del sistema
Bandera de archivo	0 si ha sido respaldado; 1 si necesita respaldarse
Bandera ASCII/binario	0 para archivo ASCII; 1 para archivo binario
Bandera de acceso aleatorio	0 para sólo acceso secuencial; 1 para acceso aleatorio
Bandera temporal	0 para normal; 1 para eliminar archivo al salir del proceso
Banderas de bloqueo	0 para desbloqueado; distinto de cero para bloqueado
Longitud de registro	Número de bytes en un registro
Posición de la llave	Desplazamiento de la llave dentro de cada registro
Longitud de la llave	Número de bytes en el campo llave
Hora de creación	Fecha y hora en que se creó el archivo
Hora del último acceso	Fecha y hora en que se accedió al archivo por última vez
Hora de la última modificación	Fecha y hora en que se modificó por última vez el archivo
Tamaño actual	Número de bytes en el archivo
Tamaño máximo	Número de bytes hasta donde puede crecer el archivo

Figura 4-4. Algunos posibles atributos de archivos.

**sol:**

El campo de longitud de registro está presente en los archivos cuyos registros se pueden realizar búsquedas mediante keys, con la información de cuánto ocupa dichos registros en memoria.

3. ¿Es absolutamente esencial la llamada al sistema **open** en UNIX? ¿Cuáles serían las consecuencias de no tenerla?

**sol:**

- El primer paso que debe ejecutar todo proceso que quiera acceder a los datos de un archivo es la llamada al sistema **open** que es esencial para reducir el overhead (sobrecarga). Sería posible deshacerse de esa llamada del sistema, pero incurre en una sobrecarga significativa.
- El propósito de la llamada "open" es buscar información de archivo incluyendo sus atributos a la memoria, para facilitar la accesibilidad en las operaciones subsiguientes. Si no se utiliza la llamada de sistema abierta, tendremos que especificar el nombre del archivo para cada operación en ese archivo. En cada operación, el sistema tendrá que obtener el correspondiente nodo i. Esto daría lugar a una sobrecarga adicional, que de otro modo podría evitarse.

4. Los sistemas que soportan archivos secuenciales siempre tienen una operación para rebobinar los archivos. ¿Los sistemas que soportan archivos de acceso aleatorio la necesitan también?

**sol:**

- No, el acceso aleatorio de los archivos no necesita la operación "rebobinar" ya que si desea volver a leer el archivo, puede acceder al byte 0.

5. Algunos sistemas operativos proporcionan una llamada al sistema **rename** para dar a un archivo un nuevo nombre. ¿Hay acaso alguna diferencia entre utilizar esta llamada para cambiar el nombre de un archivo y sólo copiar el archivo a uno nuevo con el nuevo nombre, eliminando después el archivo anterior?

**sol:**

Si. La llamada al sistema **rename** no cambia información acerca del archivo tales como fecha de creación o de última edición, a diferencia de **copy**, que realiza la operación de obtener la hora del sistema para configurar estas

variables en el archivo. Además, en caso de que el disco se encuentre sin espacio disponible, es muy probable que se produzca un error. Por otra parte, **hard links** se ven afectados por estas operaciones, mientras en un **rename** el hard link al inodo no cambia su contador, en operaciones de **copy** y **delete**, el contador del inodo del antiguo archivo disminuye.