

FAT, o un tp muy grande

Aclaraciones

- Para aprobar la totalidad del TP es necesario tener aprobado cada uno de sus módulos.
- **Fecha de entrega: Jueves 03 de Mayo de 2018**

Informe

Proponemos realizar una exploración en profundidades del sistema de archivos *FAT 12*.

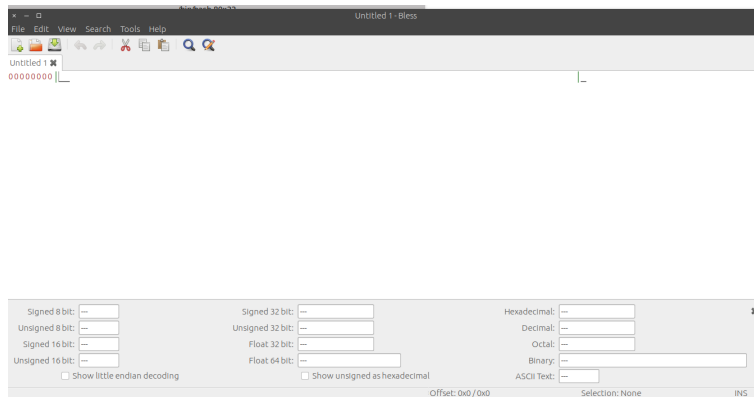
Para esto estaremos trabajando con el archivo de imagen provisto llamado *fat.img*.

Lo estaremos leyendo a bajo nivel, o sea directo del iso, pero podrán montarlo para entender y comprobar lo que van mirando.

Para esto deberán:

- Montarlo: Siendo root (o mediante sudo) ejecutar:
`mount fat.img /mnt -o loop,umask=000`
- Desmontarlo: Siendo root (o mediante sudo) ejecutar:
`umount /mnt`

Para realizar los ejercicios se deberá instalar un editor hexadecimal. Dicho editor nos permitirá leer archivos a bajo nivel. Nos permitirá ver qué hay en un byte exacto y traducirlo a varios formatos, como ser binario, hexadecimal o texto. Uno de estos editores es **bless** que se encuentra como paquete en *ubuntu*



Se deberán responder las siguientes preguntas. Todo código deberá estar correctamente documentado. A su vez deberán crear un archivo para la compilación del código completo. Genere los distintos puntos de código en archivos separados. Documente su uso (y no el código) Deberán entregar el informe por mail a la casilla atcach@ungs.edu.ar con el subj:TP-<apellido.un_integrante>. En el cuerpo del mail deberán poner nombre y documento de cada uno de los integrantes. El código y el archivo de imagen deberán estar comprimidos en un archivo con el mismo nombre del subject del mail y de extensión .tar.gz

Ejercicios:

1. Al montarlo. ¿ Para qué se ha puesto `umask=000` ?
2. Cargando el *MBR*
 - a) Muestre el *MBR* con el *Hex Editor*. Muestre los primeros bytes y la tabla de particiones. ¿Cuántas particiones hay ? Muestre claramente en qué lugar puede observarlo.
 - b) Lea los datos del punto anterior y muéstrelos por pantalla mediante por código C.
 - c) Muestre en el *Hex Editor* si la primer partición es booteable o no. ¿Lo es?
 - d) Muestre, mediante un programa en C, para la primer partición: el flag de booteable, la dirección *Cylinder-head-sector (chs)*, el tipo de partición y su tamaño en sectores.
3. Cargando la tabla de archivos
 - a) ¿Cuántos y cuáles archivos tiene el filesystem ? Muésrelos con Bless y genere el código para mostrarlos.

- b)* ¿Hay archivos borrados ? ¿Cuáles ?
- c)* ¿Qué puede decir acerca del recupero de archivos ?

4. Leyendo archivos.

- a)* Muestre, mediante el **hex editor** y mediante código **C** lo que hay en el archivo no borrado.
- b)* Modifique el filesystem para que el archivo borrado sea recuperado. (Puede hacerlo mediante el **hex editor**

Referencias:

https://en.wikipedia.org/wiki/Design_of_the_FAT_file_system

<http://www.c-jump.com/CIS24/Slides/FileSysDataStructs/FileSysDataStructs.html>