**UF2. Practica 1.** **Sistemes operatius lliures en xarxa**

**1. Introducción**

**En esta práctica veremos algunas operaciones básicas sobre archivos que nos permitirán conocer propiedades básicas de los mismos. Estas operaciones básicas deberían ser útiles para diferenciar los sistemas de ficheros propios de Linux y de Windows NT.**

**El sistema de ficheros propio de Linux es ext3 (**[**http://es.wikipedia.org/wiki/Ext3**](http://es.wikipedia.org/wiki/Ext3)**)**

**(o ext4,** [**http://es.wikipedia.org/wiki/Ext4**](http://es.wikipedia.org/wiki/Ext4) **en las últimas versiones del kernel Linux). En los sistemas operativos Windows NT el sistema de ficheros es ntfs**

**(**[**http://es.wikipedia.org/wiki/NTFS**](http://es.wikipedia.org/wiki/NTFS)**).**

**Algunas de las propiedades más relevantes de ext3 (y de ext4,** [**http://upload.wikimedia.org/wikipedia/commons/5/53/Tabla\_ext3\_ext4.png**](http://upload.wikimedia.org/wikipedia/commons/5/53/Tabla_ext3_ext4.png)**) tienen que ver con su gestión por inodos, con admitir enlaces débiles y fuertes, con el sistema de permisos que aceptan, o con el hecho de admitir (al igual que hace ntfs) “journaling”; “journaling” es una propiedad de ciertos sistemas de ficheros (y de bases de datos) por la cual los cambios hechos a un fichero no se aplican directamente sobre el mismo, sino que van a parar a un registro intermedio. En caso de que nuestro sistema falle, el sistema nos permitirá recuperar el registro de cambios realizados al mismo.**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**2. Permisos**

**Aún existen algunos otros tipos de ficheros propios en los sistemas Linux que aquí omitiremos. Pasamos ahora a ver algunas de las características del sistema de permisos propio de Linux.**

**El sistema de permisos propio de Linux se basa en definir tres tipos de permisos (lectura, escritura y ejecución, ó r-w-x) para tres tipos de usuarios (el propietario, el grupo y el resto de usuarios de la máquina, ó u-g-o).**

**De ahí que siempre nos encontremos, al listar (“ls -l”) el contenido de un directorio, con nueve letras que definen los permisos que se tienen sobre cada fichero y directorio:**

**( )rwxrwxrwx ( ) propietario grupo (datos adicionales) nombre.**

**En el caso anterior, el fichero presentado tendría todos los permisos posibles (rwxrwxrwx) para todos los usuarios de la máquina (veremos más adelante qué significa leer, escribir y ejecutar para cada tipo de fichero en Linux).**

**Los tres primeros caracteres indican los permisos para el propietario, los caracteres 4, 5 y 6 indican los permisos para los restantes miembros del grupo, y los tres últimos caracteres indican los permisos para el resto de usuarios. Si alguno de los permisos no está asignado, en su lugar aparece un guión:**

**( )rwxr-xr-- ( ) propietario grupo (datos adicionales) nombre.**

**Una breve descripción de lo que significa cada uno de los permisos sería la siguiente:**

**r:**

**Sobre ficheros: Permiso para leer un fichero.**

**Sobre directorios: Permiso para listar el contenido de un directorio.**

**w:**

**Sobre ficheros: Permiso para modificar un fichero.**

**Sobre directorios: Permiso para crear y borrar ficheros y directorios contenidos en él.**

**x:**

**Sobre ficheros: Permiso para ejecutar un fichero o script.**

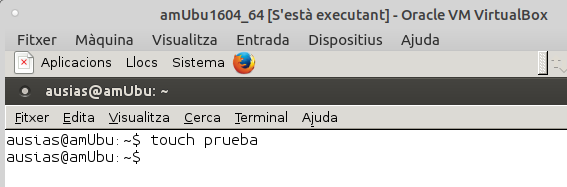
**Sobre directorios: Permiso para acceder a un directorio.**

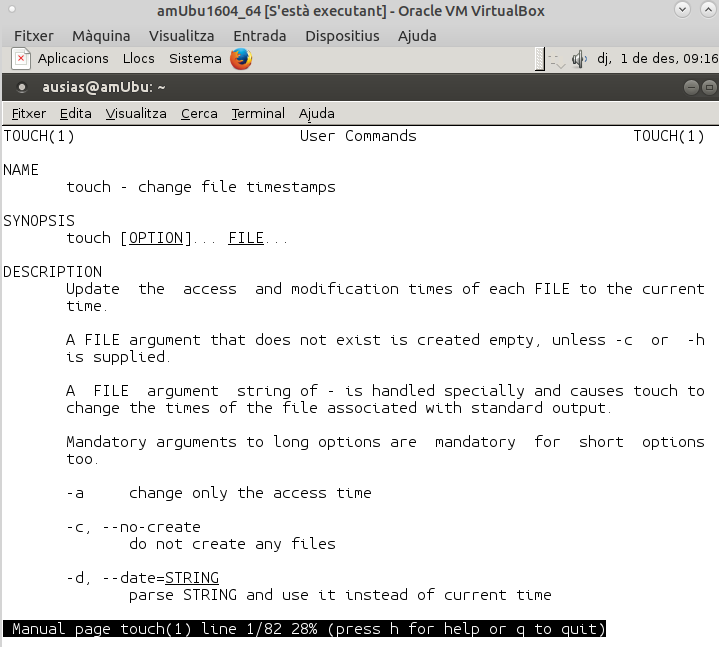
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**3. Puesta en marcha de la práctica**

**1. En primer lugar, vamos a recuperar los tipos de ficheros que nos podemos encontrar en un sistema de ficheros de una máquina Linux. Arranca la máquina virtual de Linux y dirígete al escritorio. Ejecuta el siguiente comando:**

**$touch prueba**



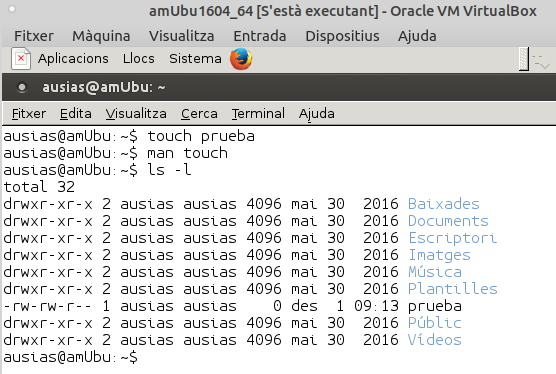
**1.1 ¿Qué ha sucedido? Comprueba el cometido del comanda “touch” por medio de “man touch” (o “info touch”).**

Al ejecutar el touch prueba ni ha sucedido nada.

Al ejecutar el comando man touch se nos abre

un manual de información.

**2. Vamos a comprobar ahora el tipo de fichero a que corresponde “prueba”. Ejecuta el comanda “ls -l”. De los 10 caracteres primeros que obtienes en su descripción (-rw-r--r--), el primero (-) corresponde con el tipo de fichero.**

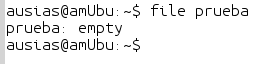


**2.1 Comprueba en la página** [**http://en.wikipedia.org/wiki/Unix\_file\_types**](http://en.wikipedia.org/wiki/Unix_file_types) **a qué tipo de**

**fichero corresponde.**

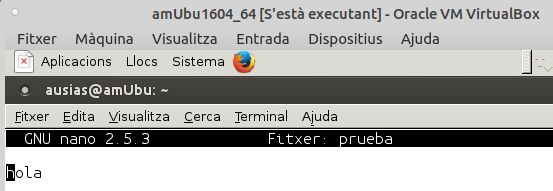
Corresponde a un fichero regular.

**3. Una orden útil en Linux que nos permite saber qué tipo de fichero tenemos delante es el mandato file. Ejecuta el mandato “file prueba”. ¿Qué información te ha facilitado sobre el fichero?**



Nos dice que el fichero está vacío.

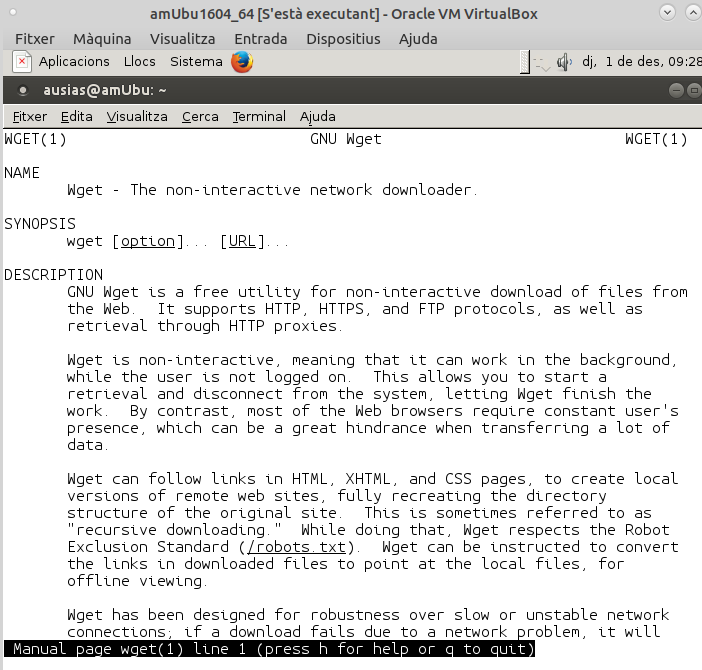
**4. Ejecuta nano sobre el fichero prueba (nano prueba). Cambia su contenido. Apunta ahora el resultado de realizar “file prueba”.**





**5. Vamos ahora a capturar una página web por medio de la orden “wget”. Explícalo.**

**(Puedes ejecutar “man wget” para saber algo más sobre esta orden).**

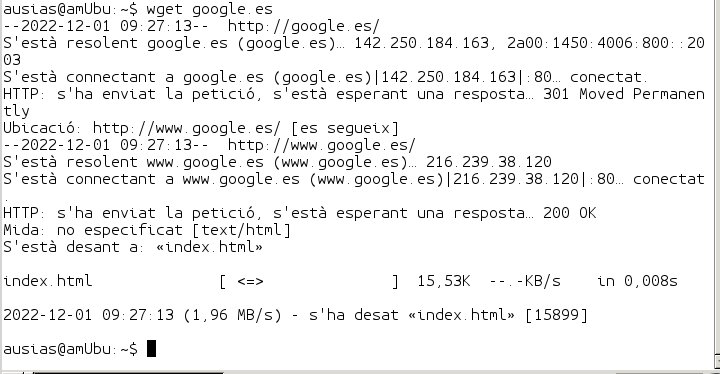


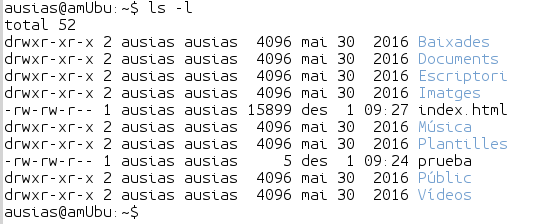
Ejecutamos wget y una url para capturar y guardar la web

**5.1 Ejecuta:**

**$wget “una página web” (sin las comillas)**

**Observa (por medio de “ls -l” o por medio del entorno de ventanas) que en tu escritorio ha aparecido un fichero de nombre “index.html” (o “index.htm”) que corresponde con la página de inicio de tu sitio web.**





**5.2 Vamos a ejecutar sobre el mismo el comando:**

**$file index.html**

**¿Qué respuesta obtienes?** información sobre el fichero



**6. El hecho de que el sistema de ficheros reconozca “index.html” como un fichero de tipo “texto de HTML” no tiene nada que ver con la extensión del mismo. Ejecuta el siguiente comando:**

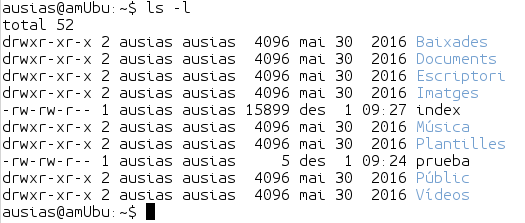


**$mv index.html index**

**¿Qué ha sucedido?**

No sucede nada.

**6.1 Compruébalo por medio de “ls -l”.**



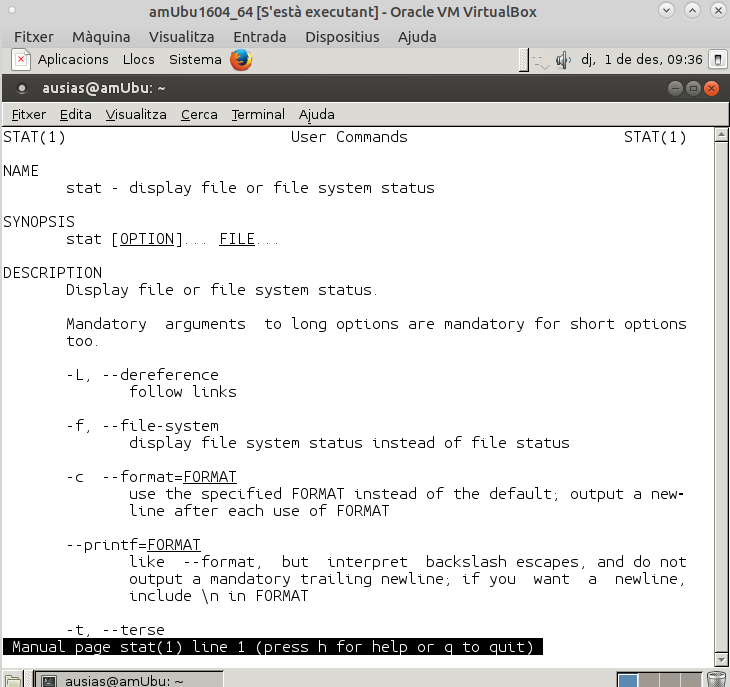
**6.2 Ejecuta ahora el mandato:**

**$file index**

**¿Qué respuesta obtienes ahora?** el fichero ya no se llama file.html ahora se llama html

**6.3 ¿Qué hubiera sucedido en un sistema Windows si hubiéramos modificado la extensión de un archivo? ¿El sistema de ficheros seguiría reconociendo con qué tipo de aplicación debería abrirlo?**

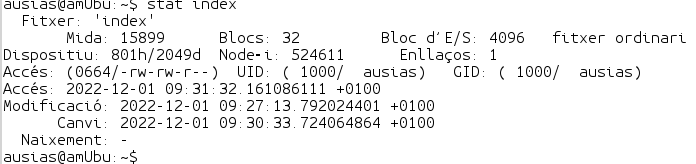
En windows el fichero quedaría inutilizable. No lo reconocería.

**7. Otro comando Linux que también nos provee de información interesante sobre un archivo o carpeta es “stat”. Ejecuta “man stat” para saber qué función realiza.**

**7.1 Ejecuta y explica los comandos:**

**$stat index: Su función es mostrar el estado**

**del archivo o del sistema.**



**$ls –li index: Nos dice el tipo de fichero que tiene.**



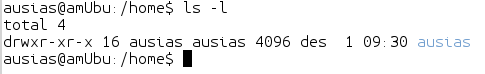
**7.2 Observa los distintos datos que nos facilitan ambos mandatos sobre el fichero. ¿Cuáles son comunes? ¿Qué información guarda el sistema de ficheros sobre cada fichero?**

Guardan información sobre cuándo se ha modificado el fichero por última vez, el tamaño, el acceso, el tipo de fichero, etc.

**8. Otro elemento típico del sistema de ficheros de Linux son los directorios. En Linux los directorios están implementados de manera interna como archivos que contienen listas de archivos. Sitúate en tu carpeta de inicio (/home/alumno o directamente~). Ejecuta y explica el comando:**

**$ls –l**

No existe el /alumno por tanto no deja acceder.



Nos dice el directorio que hay en esa ubicación, el tipo y la hora.

**8.1 ¿Qué tipo de fichero es Escritorio (cuál es la primera letra de su descripción)? Comprueba su significado en** [**http://en.wikipedia.org/wiki/Unix\_file\_types**](http://en.wikipedia.org/wiki/Unix_file_types)**.**



Directory: es el archivo más común, se define por el sistema de archivos utilizados. La primera letra de su descripción es la d.

**8.2 Ejecuta y explica también el comando:**

**$file Escritorio**

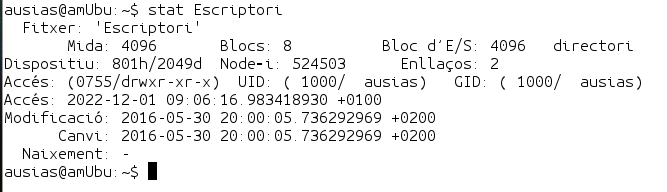


Nos dice el tipo de archivo

**¿Qué respuesta has obtenido?** Que el fichero Escritorio es Directory

**8.3 Por último, ejecuta el comando:**

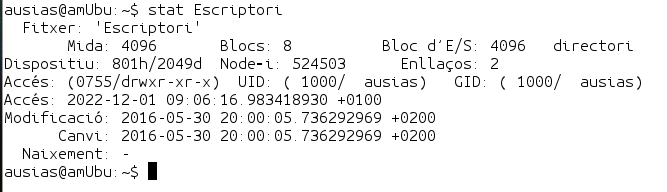
**$stat Escritorio**



**Apunta el número de enlaces (fuertes) que existen a Escritorio.**

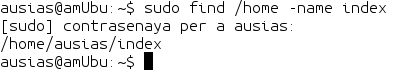
Existen 2 enlaces.

**9. Una aplicación que puede ser útil para encontrar ficheros en los sistemas Linux es “find”. Puedes ejecutar “man find” para aprender algo más sobre la misma. El mandato “find” debe tener privilegios para poder acceder a las distintas carpetas de nuestro sistema.**



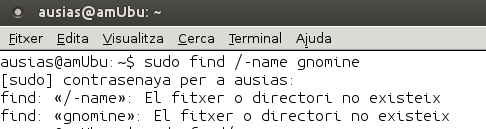
**9.1 Si quieres buscar un fichero en todo el sistema, deberías usarlo con “sudo”. Si sólo quieres buscar un fichero en tu directorio personal, lo anterior no es necesario. Ejecuta y explica los comandos:**

**$find /home/alumno –name index**



**9.2 ¿Qué salida has obtenido? La dirección del fichero**

**$sudo find / -name gnomine**



**9.3 ¿A partir de qué directorio se ha producido la búsqueda?** Del directorio raíz /

**10. Los dos tipos de ficheros anteriores (ficheros normales y directorios) no son los únicos que podemos encontrar en un sistema Linux. Desde el Escritorio crea un enlace débil al fichero “/usr/games/gnomine”.**

**$ln –s /usr/games/gnomine enl\_debil\_gnomine**

**10.1 En el mismo Escritorio ejecuta ahora el comando:**

**$file enl\_debil\_gnomine**

**(Apunta el resultado obtenido)**



**10.2 En el mismo Escritorio ejecuta ahora el comando:**

**$ls –l**

**¿Qué tipo de fichero es?(cuál es el primer carácter) ¿es el mismo?**

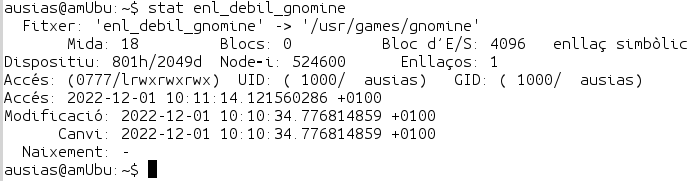
**(**[**http://en.wikipedia.org/wiki/Unix\_file\_types**](http://en.wikipedia.org/wiki/Unix_file_types)**)**

Es un symbolic link, el primer carácter és l, no es el mismo.

**10.3 En el mismo Escritorio ejecuta ahora el comando:**

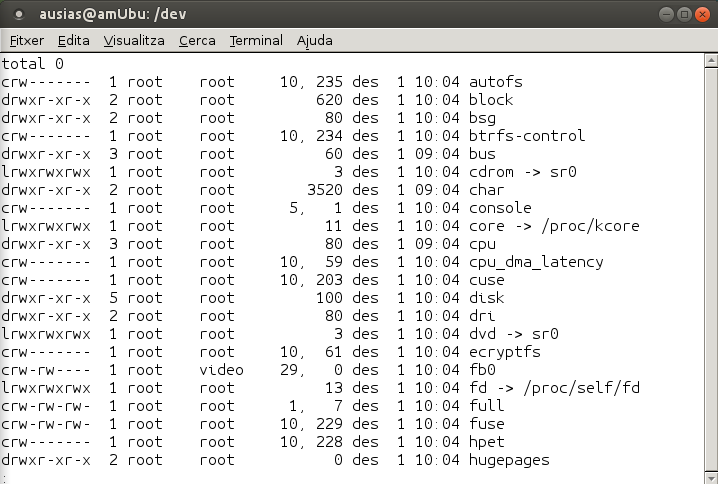
**$stat enl\_debil\_gnomine**

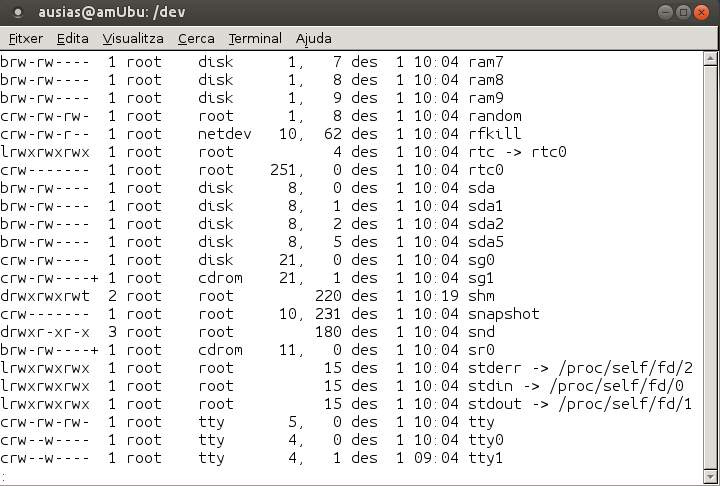
**Localiza la información referente a los permisos del fichero.**

Los permisos són: lrwxrwxrwx

**11. Otros dos tipos de ficheros bastante comunes en los sistemas Linux son los dispositivos de bloques y de caracteres. Los mismos se utilizan para representar algunos dispositivos hardware tales como discos duros o tarjetas de sonido, o también las propias terminales del sistema.**

**Dirígete a la carpeta “/dev”. Ejecuta el comando:**

**$ls –l | less**



**11.1 Apunta algunos de los dispositivos cuyo nombre reconozcas (“tty...” corresponde con terminales, “sd...” corresponde con los dispositivos de almacenamiento... etc.)**

**(**[**http://en.wikipedia.org/wiki//dev/sda#Naming\_conventions**](http://en.wikipedia.org/wiki/dev/sda#Naming_conventions)**).**

ram0

sda

sg0

net

tty0

console

full

disk

**11.2 ¿A qué tipo de fichero dentro de los propios de Linux corresponden?**

**(**[**http://en.wikipedia.org/wiki/Unix\_file\_types#Device\_file**](http://en.wikipedia.org/wiki/Unix_file_types#Device_file)**).**

ram0:device file

sda:device file

sg0:device file

net:mode string

tty0:device file

console: device file

full: device file

disk: mode string

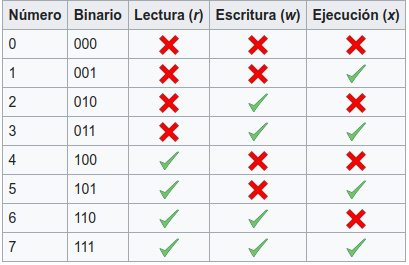
**12. Un comando que permite modificar los permisos de un fichero o directorio es “chmod”. Puedes consultar su manual (“man chmod”). Los permisos se pueden modificar tanto por medio de caracteres como por el modo octal. Por el modo de caracteres, podemos especificar los permisos para el propietario (u), su grupo (g) o el resto de usuarios (o), y los permisos serán de la forma +rwx, -rwx ó =rwx. Por medio de + y – añadimos o eliminamos permisos, por medio de = asignamos unos nuevos.**

**En el modo octal la notación es un poco distinta. El permiso de lectura se asocia con un 4, el de escritura con un 2 y el de ejecución con un 1. La asignación de permisos sale de la suma de los anteriores números.**

**Por ejemplo, 6 es permiso de lectura y escritura, pero no de ejecución. 7 es igual a todos lo permisos, y 0 es igual a ninguno. Por medio de ternas “657”, asignamos los permisos del usuario, su grupo y el resto de usuarios respectivamente.**

**Puedes encontrar ayuda en la web:**

[**http://es.wikipedia.org/wiki/Chmod#Asignaci.C3.B3n\_de\_permisos\_en\_el\_comando\_chmod**](http://es.wikipedia.org/wiki/Chmod#Asignaci.C3.B3n_de_permisos_en_el_comando_chmod) **una descripción más detallada. Explícala.**



Mediante el terminal podemos crear ficheros y con el comando chmod modificar sus permisos, por ejemplo:

chmod 766 file.txt

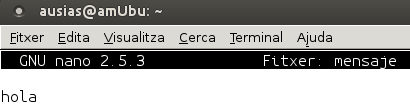
Con este comando assignemos permisos de control total al propietario y al resto solo de lectura y escritura.

**13. Muévete al escritorio. Crea un fichero “mensaje” y escribe en el mismo un texto sencillo. Crea también tres ficheros de texto f1, f2 y f3 con el contenido que quieras. Comprueba con “ls -l” los permisos que han sido asignados a cada uno de los ficheros.**

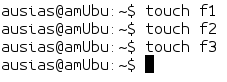
Creamos el fichero mensaje.



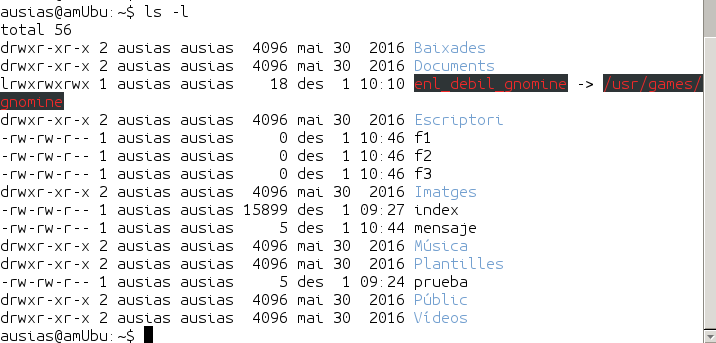
Hacemos sudo nano mensaje y ponemos un texto sencillo.



Creamos los otros ficheros



Miramos los permisos de los ficheros



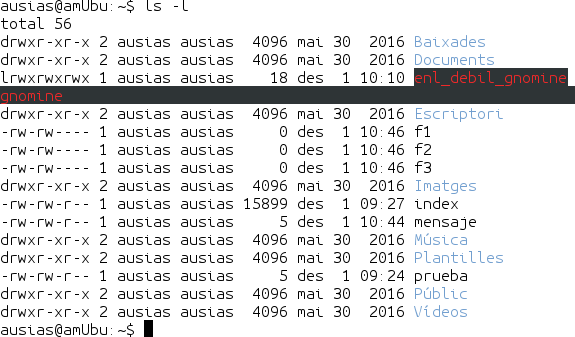
**14. Deniega al “resto de usuarios” todos los permisos sobre f1, f2 y f3.**

**$chmod o-rwx f1 f2 f3**

**(El carácter “-” se utiliza para quitar los permisos, el carácter “+” para asignarlos).**

****

**14.1 Comprueba con “ls -l” los cambios ocurridos.**

****

**15. Deniega a todos los usuarios los permisos de lectura y ejecución de f1.**

**$chmod ugo-rx f1 (también es posible chmod a-rx f1)**

****

**15.1 Comprueba con “ls -l” que el cambio ha tenido lugar. Intenta acceder al fichero (por ejemplo con less). Apunta el resultado.**

