

Introducción al sistema operativo GNU/Linux

1.- ¿Que es Linux?

Linux es un sistema operativo de gran difusión y de libre distribución. Permite trabajar en modo monopuesto o en modo red, permitiendo al usuario iniciar sesión localmente o en forma remota. Se puede instalar en gran variedad de plataformas hardware.

Componentes principales

- Shell: Sirve para introducir ordenes, ejecutar programas.
- Nucleo: Es la parte que interactúa con el hardware.

2.- El shell de Linux

Interactuamos con el sistema a través del terminal, al terminal de Linux lo llamamos Shell. Hay diferentes distribuciones.

- Shell Bourne (sh): El más utilizado.
- C-Shell (csh): Ofrece grandes características para programadores en C
- Shell Korn (KSh): Con características ampliadas de programación.
- Bourne Again Shell (Bash): Incorporado con Ubuntu, compatible con el shell sh.

3.- Ordenes básicas de Linux

Las ordenes son programas que están almacenados en el sistema de archivos. Al principio Linux usaba la ejecución de ordenes en entorno de comandos. Emplean el siguiente formato:

- Orden: Acción a realizar
- Opciones: Se introducen seguidas del carácter (-)
- Argumentos: Puede ser un archivo, documento u otra información.

4.- Directorios y carpetas de Linux

Un directorio es un tipo de archivo que almacena información a cerca de los archivos y directorios que contiene. Tienen características que los clasifican y ubican en un sistema de archivos.

Pueden emplear casi cualquier carácter a excepción de la " / ". Difiere entre mayúsculas y minúsculas también admite nombres largos.

Las operaciones con carpetas son las mismas que en Windows.

La estructura de directorios y archivos emplea una jerarquía en árbol. No hay límite de archivos y directorios a crear dentro de un directorio.

5.- Sistema de archivos en Linux

Es el modo en que SO organiza los archivos del dispositivo de almacenamiento. Linux posee una estructura jerárquica en forma de árbol. El directorio principal es la raíz "/" y desde ella crece toda la estructura del sistema.

Características del sistema de archivos

El sistema de archivos permite al usuario crear, borrar y acceder a los archivos sin necesidad de saber el lugar exacto en el que se encuentran.

No tiene unidades físicas, sino ficheros que hacen referencias a ellas.

El sistema de archivos consta de varias partes importantes:

- Bloque de carga: Está reservado para almacenar un programa que utiliza el sistema para gestionar el resto de las partes del SO
- Superbloque: Contiene información sobre el sistema de archivos.
- Tabla de i-nodos: Cada archivo tiene asociado un elemento en esta tabla con un número que identifica la ubicación del archivo dentro del área de datos.
- Área de datos: Ocupa el resto del disco y es donde se almacenan ficheros y directorios.

En Linux cada bloque tiene 512KB o múltiplos de 512. En el sistema de ficheros ext4 la parte más importante es el bloque.

Tipos de sistemas de archivos en Linux

Linux soporta una gran variedad de sistema de ficheros:

- FAT32: Sistema tradicional de MS-DOS
- NTFS: Usado por Windows, Linux solo lo recomienda para lectura por ser privativo.
- ext3: Versión mejorada de ext2 con previsión de perdida de datos por fallo del disco por apagones. Imposible recuperar datos borrados.
- ext4: Implementa mejoras respecto a ext3:
 - Mayor tasa de transferencia y mejor rendimiento
 - Sistema de ficheros de mayor tamaño.
 - Más de 32000 subdirectorios.
 - Apenas se fragmenta, pero añade soporte para fragmentación.
 - Posee herramienta para recuperar archivos borrados. (undelete)
- ReiserFS: Gestiona archivos pequeños eficientemente y permite journaling.
- swap: Sistema de archivos para la zona de intercambio. Partición empleada para cargar programas y no saturar la memoria ram.

Directorios más importantes de Linux

Directorio raíz (/): De le parten todos los directorios del sistema. Los más importantes son:

- /bin: Contiene comandos y ficheros lanzables por los usuarios.
- /sbin: Con archivos de configuración del sistema.
- /home: Contiene todos los directorios personales de los miembros del sistema.
- /usr: Contiene utilidades y programas de uso general para los usuarios.
- /dev: Contiene archivos de bloques y caracteres asociados a dispositivos de hardware
- /lib: Contiene librerías y compiladores del sistema.
- /etc: Directorio con ficheros de configuración y utilidades del sistema.
- /media: Contiene todas las unidades físicas que tenemos montadas en el sistema.

Trayectorias en Linux

Los caminos, rutas o trayectorias se dividen en absolutas y relativas. No existe la referencia a ninguna unidad lógica. Las rutas absolutas van desde la raíz hasta el propio archivo o directorio.

Metacaracteres en Linux

Conjunto de caracteres con significado especial para búsqueda o operaciones con archivos.

6.- Ayuda en Linux

Usado en entorno de comandos para obtener ayuda sobre los comandos. Dos formas...

- Teclear el comando **man** seguido de un espacio y el comando a consultar
- Teclear el comando seguido de un espacio y **--help** para obtener la ayuda.

7.- Operaciones sobre directorios, carpetas y permisos

Estas operaciones son muy similares a las de Windows en entorno gráfico. En entorno de comando pueden variar en cuanto a las ordenes, pero siguen siendo muy parecidas.

8.- Atributos de los directorios o carpetas

En Linux cada archivo queda identificado por 10 caracteres

A los 10 caracteres que acompañan a cada archivo y directorio de Linux se le denomina Máscara. Esta *Máscara* representa los permisos que tienen los usuarios del sistema.

- - : Representa el tipo de archivo
- r : Permiso de lectura, permite saber que contiene.
- w: Permiso de escritura, permite crear archivos en el directorio
- x : Permiso de ejecución, permite ejecutar el directorio y copiar archivos de el

Exceptuando el primer carácter, cada grupo de tres caracteres hacen referencia al tipo de operaciones que se pueden realizar por cada tipo de usuario.

- Primer carácter: (-) Archivo normal, (d) directorio, (l) Archivo de vinculo
- Primer grupo de 3 caracteres: (u) Propietario
- Segundo grupo de 3 caracteres: (g) Grupo al que pertenece el usuario.
- Tercer grupo de 3 caracteres: (o) Otros usuarios

Premisos de directorios, la máscara

Un archivo o directorio de Linux es del usuario que lo creó, el usuario pertenece a un grupo de usuarios que pueden usar el archivo o directorio y el archivo o directorio puede ser usado por otros usuarios que no pertenecen al grupo de usuarios ni son el usuario.

En la creación de directorios empleamos una máscara octal que es:

- 777 – 022 (valor de umask por defecto)= 755
- Pasado a binario: 111 101 101
- Que equivale a los permisos: rwx r-x r-x.

El administrador del sistema tiene todos los privilegios. El resto de usuarios puede hacer lo que quiera con sus archivos o carpetas, con el resto que no le pertenezcan solo podrá hacer lo que el administrador le tenga asignado.

Con el comando **chmod** podemos quitar derechos a cada tipo de usuario. Si no especificamos al usuario se añaden o quitan permisos a todos los usuarios.

Con el comando **umask** desde el momento de ser creado cualquier archivo o carpeta, tendrá los valores que se le asignen con este comando. Los valores por defecto son modificados.

Introducción al sistema operativo GNU/Linux – Parte II

1.- Introducción a los archivos

El manejo de los archivos en entorno gráfico es similar a otros sistemas operativos y se pueden realizar prácticamente las mismas operaciones con las carpetas y directorios.

Linux es “*case sensitive*” (diferencia mayúsculas y minúsculas) con el nombre de los archivos. La extensión no es obligatoria.

2.- Tipos de archivos en Linux

Existen 5 tipos de archivos:

- Ordinarios: Contienen información con la que trabaja el usuario.
- Enlaces físicos: Dan acceso al archivo original, permiten modificarlo y trabajar con él.
- Enlaces simbólicos: Accesos directos a un archivo.
- Directorios: Archivos especiales que contienen otros archivos.
- Archivos especiales: Representan dispositivos físicos, se tratan como archivos ordinarios.

3.- Operaciones con archivos

En entorno gráfico es similar a otros sistemas operativos.

En entorno de comandos tenemos determinadas ordenes:

- cat: Permite ver los archivos
- mv: Mueve archivos, puede renombrarlos
- cp: Copia archivos.
- rm: Elimina archivos.

4.- Permisos y atributos en los archivos

Funcionan igual que en los directorios, salvo que por defecto los permisos efectivos en la creación de ficheros son 666 y para calcular sus permisos hay que:

- Pasar el umask deseado (022) a binario = 000 010 010
- Realizar la negación = 111 101 101
- Multiplicar la negación por el valor base de los ficheros (666) en binario 110 110 110
 - $111\ 101\ 101 \times 110\ 110\ 110 = 110\ 100\ 100 \rightarrow 644$ en octal
- Que equivale a los permisos: rw- r-- r--.

5.- Compresión de archivos

En Linux se puede comprimir archivos o carpetas con los siguientes formatos:

- tar: Solo para directorios
- zip, rar, gz, bz2: Para archivos o directorios

6.- Edición de textos en Linux

Para trabajar con archivos de texto plano podemos emplear los siguientes editores de texto.

- vi: Antiguo y complicado de usar, solo con teclado y sin teclas de edición
- nano: Similar a vi, menos complicado.
- Gedit: Ejecutable desde línea de comandos y terminal

7.- Órdenes básicas

- `clear`: Limpia la pantalla del terminal
- `date`: Nos muestra la fecha y la hora
- `cal`: Enseña un calendario
- `who`: Muestra los usuarios conectados y algunas datos adicionales
- `logname`: Muestra el contenido de esta variable
- `uname`: Muestra información del tipo de sistema
- `id`: Muestra número de identificación y el grupo del usuario
- `passwd`: Permite cambiar o asignar una contraseña.

8.- Comprobación de dispositivos de almacenamiento

En Linux la comunicación entre el sistema y los dispositivos físicos de almacenamiento de datos se realiza mediante archivos. Estos son similares a otros archivos del sistema.

Estos archivos de dispositivos se encuentran en la carpeta */dev*, por ejemplo:

- `sda`: pertenece al disco duro principal
 - `sdb, sdc...` : Pertenecen al resto de discos duros o unidades usb externas.
 - En las particiones de cada unidad añadimos un número: `sdb0, sdb1...`
- `scd0, scd1...` : Para unidades de dvd / cd-rom
- `fd0, fd1...` : Para unidades de disquetes

Montaje y desmontaje de dispositivos

Para poder trabajar con dispositivos de almacenamiento primero tenemos que montarlos en el sistema. Esto consiste en añadirlo a la estructura de archivos del sistema.

El *enlace* consiste en indicar en que punto del sistema de archivos colocamos la raíz del sistema de archivos del dispositivo montado.

Con las ordenes ***mount*** y ***unmount*** montamos y desmontamos dispositivos de almacenamiento al sistema principal de archivos.

Un dispositivo no debe de estar montado en más de un sitio a la vez.

Dar formato a dispositivos de almacenamiento

Empleamos el comando ***mkfs*** y el dispositivo debe de estar desmontado

Manejo de discos en Linux

El comando ***fsck*** sirve para recuperar archivos mal referenciados por la tabla de i-nodos.

9.- Gestionar el hardware del equipo

La mayoría del hardware del mercado es detectado e instalado automáticamente.

Los controladores suministrados por los fabricantes de hardware se llaman controladores privativos, el resto de controladores son genéricos y están suministrados por el SO.

Gestión de usuarios y grupos en GNU/Linux

1.- Introducción

Linux es un SO multiusuario por lo que puede tener a varios usuarios trabajando en el sistema de forma simultánea. Deben pertenecer a uno o varios grupos.

2.- Usuarios y grupos

Para entrar en Linux debemos identificarnos como usuario. Estos se identifican con un número de usuario llamado UID (user ID).

Cada usuario debe pertenecer a un grupo llamado grupo principal, además puede pertenecer a otros grupos llamados secundarios.

Un grupo es un conjunto de usuarios con su identificador de grupo GID (group ID)

Tipos de usuarios

- Usuario root: Superusuario o administrador
 - Cuenta con todos los privilegios.
 - Tiene acceso total.
 - Su UID es 0
- Usuario del sistema:
 - Se generan al instalar el sistema o cualquier servicio.
 - Dependiendo del usuario tiene unos u otros privilegios.
 - Su UID es mayor de 1 y menor de 1000
 - El usuario especial *nobody* tiene el ultimo UID posible 65534
- Usuarios normales:
 - Se conectan al sistema introduciendo su clave y su contraseña
 - Tienen un directorio de trabajo dentro de /home
 - El UID se genera a partir del 1000

Ficheros de configuración

Son los ficheros que el sistema lee o modifica cuando gestiona los usuarios y los grupos. La información está separada por el carácter “ : ”

- /etc/passwd: Contiene líneas información sobre cada usuario. Tiene la estructura...

login:x:UID:GID:información:directorio_personal:shell_de_inicio

- login: Nombre del usuario con el que se identifica al entrar al sistema.
- x: posee una contraseña almacenada y encriptada en el fichero /etc/shadow
- UID: Identificador de usuario, entero igual o superior a 0
- GID: Identificador de grupo, entero igual o superior a 0
- Información: Información sobre el usuario.
 - Conocido como GECOS
 - Separada por comas
 - Visualizable con el comando ***finger***
- directorio_personal: Ruta absoluta del directorio personal del usuario
- shell_de_inicio: Shell de entrada por defecto del usuario

- `/etc/shadow`: Almacena las contraseñas de los grupos del sistema. Solo root puede leerlo
- nombre:contraseña:administradores:miembros
- nombre: Nombre del grupo
 - contraseña: Puede ser una contraseña encriptada o los caracteres * ó !

Otros ficheros de configuración

- `/etc/default/useradd`: Contiene valores por defecto al añadir usuarios con `useradd`
- `/etc/adduser.conf`: Contiene valores por defecto al añadir usuarios con `adduser`
- `/etc/deluser.conf`: Contiene valores por defecto al eliminar usuarios con `deluser`
- `/etc/login.defs`: El archivo define algunos valores por defecto sobre la encriptación de contraseñas y otros parámetros al generar un usuario nuevo.
- `/etc/shells`: Contiene la lista de shells validos para el usuario.
- `/etc/skel`: Tiene el contenido del directorio de los nuevos usuarios.

3.- Comandos para la gestión de usuarios y grupos

su – Cambia de usuario o permite ser el superusuario (root)

- `su -usuario`
 - Si no indicamos usuario asume que el usuario es root

sudo - Permite ejecutar comandos como si fuéramos el root

- `sudo comando`
 - pide contraseña de root como medida de seguridad.

adduser – Añade usuarios al sistema, o si existen los añade a un grupo que exista.

- `adduser (opciones) usuario`
- `adduser (opciones) usuario grupo`
 - `--ingroup nombreGrupo`: crea al usuario con el *nombreGrupo* como grupo principal
 - `--gid numero gid`: Como anterior pero usa el GID
 - `--home directorio`: hace que *directorio* sea el directorio personal en vez de `/home`

deluser – Elimina un usuario.

- `Deluser (opciones) usuario`
 - `--remove-home`: borra el subdirectorio personal del usuario
 - `--remove-all-files`: Elimina su directorio y todos sus archivos.

usermod – Modifica un usuario ya creado.

- `usermod (opciones) usuario`
 - `-d` : cambia el directorio personal
 - `-g grupo` : Cambia el grupo primario del usuario.
 - `-l` : cambia el nombre del usuario
 - `-s shell` : Cambia el shell del usuario

addgroup – Añade un grupo al sistema.

- `Addgroup nombreGrupo`.

delgroup – Elimina un grupo. No se puede borrar si tiene un usuario asignado.

- `delgroup grupo`

groupmod – Modifica un grupo ya creado.

- `Groupmod (opciones) grupo`
 - `-n nombre` : Cambia el nombre del grupo
 - `-g GID` : Cambia el GID del grupo

gpasswd – Añade o elimina usuario de un grupo. Añade contraseñas a un grupo.

- `gpasswd (opciones) usuario grupo`
- `gpasswd grupo` (para añadir contraseña al grupo)
 - `-a` : añade el usuario al grupo
 - `-d` : elimina al usuario del grupo
 - `-M` : añade vario usuarios al grupo.

id – Muestra información sobre el *usuario*

- id (opciones) usuario
 - -u : Escribe solo el UID del usuario
 - -n : Escribe el nombre del usuario.

groups – Muestra el nombre de todos los grupos a los que pertenece el usuario.

- groups *usuario*

finger – Muestra información del usuario que se le indique.

- finger *usuario*

Gestión del procesos

1.- Concepto de proceso

Un proceso es un programa en ejecución, emplea la memoria principal, el procesador y sus registros. Es una entidad dinámica, ya que atraviesa por diferentes estados.

Los SO multitarea pueden trabajar con varios procesos simultáneamente...

- Estableciendo prioridades de ejecución
- Asignando recursos
- Impidiendo interferencias entre procesos
- Permitiendo comunicación y colaboración entre procesos

Existen dos tipos de procesos, de usuario y de SO

Al arrancar el PC se realizan procesos del propio SO

El procesador puede trabajar en dos modos:

- Modo kernel: Los procesos tienen total libertad de uso de los recursos del sistema.
- Modo Usuario: Los procesos del usuario tienen ciertas restricciones sobre los recursos.

Cada proceso tiene un PID (Process ID) que identifica un proceso de otro.

2.- Gestión de procesos

Bloque de control de procesos (BCP)

Es un proceso del SO al que se le asocia una estructura de datos que contienen:

- Estado actual del proceso
 - En ejecución: Está usando la CPU
 - Bloqueado: Debido a que la ejecución no puede continuar.
 - Preparado: Se puede continuar con el proceso cuando la CPU esté libre.
- Identificación del proceso
- Prioridad del proceso
- Ubicación en memoria
- Recursos utilizados

Cambio de contexto: Quitar un proceso de la CPU para poner otro, necesita dos acciones:

- Guardar en el BCP del proceso los valores de los elementos del sistema.
- Restaurar el estado de los curso del sistema para el proceso siguiente.

El planificador determina el orden de los procesos preparados a ejecutar

El despachador pone en ejecución los procesos seleccionados por el planificador.

3.- Ejecución de la CPU

El modo de repartir los recursos de la CPU se determina mediante un algoritmo de planificación, que emplea los criterios siguientes:

- Equidad: Respecto al reparto del procesador
- Eficiencia: Que nunca se quede parado
- Tiempo de respuesta y tiempo de procesamiento global: Que se minimicen
- Rendimiento: Que se maximice

Mediante el uso del reloj se controla el tiempo de ejecución de los procesos.

4.- Comandos para la administración de procesos

ps – Informa sobre el estado de los procesos

- ps (opciones)
 - -A, -ely, aux : Muestras diferentes campos

pstree – Muestra procesos en forma de árbol

- pstree (opciones)
 - -A, -G, -u : Aparece con diferentes formatos

& – permite ejecutar procesos en segundo plano.

- [comando] &

kill – Termina el proceso que la indiquemos.

- kill (opciones) PID
 - -9 : termina con el proceso de la PID que indiquemos.

time – Muestra el tiempo que tarda en ejecutarse un comando

- time [comando]

top – Muestra y actualiza los procesos cada tres segundos.

- top (salimos con la tecla Q)

Comandos para la gestión de archivos / directorios en GNU/Linux

1.- Caracteres comodines

Substituyen a un carácter o a un conjunto de caracteres.

***** – Substituye a una cadena de caracteres de cualquier tamaño

? – Substituye a un único carácter

[] – Dentro puede haber un conjunto o rango de caracteres.

{ } – Dentro puede haber una cadena o varias cadenas de caracteres separadas por comas

2.- Caracteres de escape

Algunos caracteres no son imprimibles y pueden dar errores, en estos casos se incluyen estos caracteres poniéndolos entre comillas (" ") o bien la contrabarra (\)

3.- Comandos empleados

ls – Muestra información sobre ficheros y directorios

- ls (opciones) [directorio/archivo]
 - -l : Muestra la información en formato largo, muestra los permisos
 - -h : Muestra los ficheros con unidades de medida adecuadas, usar con "l" o "s"
 - -i : Muestra el i-nodo del fichero
 - -n : como -l pero muestra el GID y el UID en vez de los nombres
 - -R : Muestra la información de forma recursiva
 - -s : Muestra el tamaño de los bloques de cada fichero
 - -S : Ordena los ficheros por tamaño

pwd – Muestra la ruta absoluta del directorio donde nos encontramos

- pwd

mkdir – Crea directorios

- mkdir (opciones) *ruta y nombre directorio*
 - No ha opciones interesantes, pero haberlas ahylas.

cd – Cambia de directorio

- **cd** (opciones) ó (*ruta del directorio*)
 - - : Cambia al directorio anterior al actual
 - .. : Cambia al directorio padre
 - ~ : Nos lleva a nuestro directorio home

rmdir – Borra solo directorios si están vacíos

- **rmdir** *nombre directorio*

rm – Borra ficheros y directorios

- **rm** (opciones) [*directorio/archivo*]
 - -d : Borra el directorio aunque no esté vacío
 - -r : Borra el directorio aunque no esté vacío de forma recursiva.

cp – Copia uno o varios archivos en otro fichero o directorio

- **cp** (opciones) *fichero/directorio destino*
 - -f : Fuerza la copia
 - -a : Muestra archivos ocultos
 - -r : Copia directorios y archivos de forma recursiva

mv – Mueve un fichero a otro directorio/fichero. Usado para renombrar

- **mv** (opciones) *fichero destino*
 - -i : Pregunta antes de sobrescribir
 - -f : fuerza la sobrescritura
 - -u : Mueve solo si el destino no existe

file – Muestra el tipo de fichero

- **file** *nombre de fichero/directorio*

du – Muestra el espacio que ocupa el fichero o directorio

- **du** (opciones) *nombre fichero/directorio*
 - -h : muestra la informacion en unidades de medida entendibles

df – Muestra el espacio libre de los dispositivos de almacenamiento.

- **df** (opciones)
 - -h : muestra la informacion en unidades de medida entendibles
 - -a : Muestra hasta los archivos de tamaño 0

cat – Muestra el contenido de los ficheros, también puede crearlos mediante redireccionamientos

- **cat** (opciones) *fichero*
 - -n : muestra todas las lineas
 - > : Redirige la salida a un fichero (se abandona con Ctrl+d)

head – Muestra las 10 primeras filas de un fichero

- **head** (opciones) *fichero*
 - -n : si adjuntamos un número nos muestra esa cantidad de líneas

tail – Muestra las 10 últimas filas de un fichero

- **tail** (opciones) *fichero*
 - -n : si adjuntamos un número nos muestra esa cantidad de líneas

wc – Muestra el número del tipo de opción que se le indique.

- **wc** (opciones) *fichero*
 - -c : muestra el número de bytes
 - -m : Muestra el número de caracteres
 - -l : muestra el número de líneas
 - -w : Muestra el número de palabras
 - -L : Muestra la línea más larga

more – Muestra el contenido de forma paginada

- **more** *fichero*

less – Muestra el contenido de forma paginada, se pueden usar las teclas de dirección

- **less** *fichero*

sort – Muestra en orden ascendente un fichero. Se pueden concatenar archivos redireccionandolos

- **sort** (opciones) *fichero*
 - -c : Comprueba si está ordenado
 - -r : Ordena en sentido inverso
 - -f : Se vuelve case insensitive
 - -n : Ordena de forma numérica
 - -u : Elimina líneas repetidas
 - -t : Usa el carácter especificado entre comillas como delimitador
 - -k : Selecciona los campos indicados (uno, un rango, varios separados por comas)

ln – Crea un enlace físico al archivo/directorio indicado. A un directorio el enlace será simbólico.

- **ln** (opciones) *nombre fichero/directorio*
 - -s : crea un enlace simbólico
 - -t : especifica donde se van a crear los enlaces

cut – Muestra solo ciertas líneas verticales de los ficheros

- **cut** (opciones) *ficheros*
 - -d : Usa el carácter especificado entre comillas como delimitador
 - -f : Muestra los campos indicados (uno, un rango, varios separados por comas)
 - -c : Muestra los caracteres que se especifiquen de cada columna.

grep – Devuelve todas las líneas que contienen un patrón (texto) determinado.

- **grep** (opciones) [patrón] *archivo*
 - -i : se vuelve case insensitive
 - -n : muestra el numero de cada línea que contiene el patrón
 - -c : Muestra la cantidad de lineas que contienen el patrón
 - -r : busca en un directorio de manera recursiva
 - -l : Muestra solo el nombre de los ficheros donde encontró el patrón
 - -v : Muestra las líneas que no contienen el patrón
 - patrón: Muestra lineas que contengan la cadena patrón
 - ^patrón : Muestra lineas que empiezan por el patrón
 - ^[^patrón] : Muestra lineas que no empiezan por el patrón
 - patrón\$: Muestra lineas que terminan por el patrón
 - ^[patrón\$] : Muestra lineas que no terminan por el patrón

locate – Busca archivos dentro del sistema de archivos.

- **locate** *fichero*

find – Busca de forma recursiva todos los ficheros que cumplan unas condiciones.

- **find** [ruta] (opciones)
 - -name <expresión> : especifica patrones para los ficheros a buscar
 - -iname <expresión> : especifica patrones para los ficheros a buscar (case insensitive)
 - -type : tipo de fichero a buscar (d directorio, f archivo)
 - -size ± <n> : Tamaño máximo a buscar
 - -perm ± <modo> : especifica los permisos a buscar

diff – Permite comparar ficheros de texto linea a linea

- **diff** (opciones) *fichero-desde fichero-hasta*
 - -r : Busca de forma recursiva
 - -u : Muestra la diferencia en tres lineas

uniq – Elimina las líneas duplicadas de un fichero ordenado

- **sort** *archivo* | **uniq** (opciones)
 - -c : Genera un informe de salida precedida por un contador de líneas repetidas
 - -i : Case insensitive