

5.-Comandos para la manipulación y gestión de directorios

Comando pwd

Sintaxis:

```
pwd
```

Muestra el nombre del directorio activo.

Comando cd

Cambia el directorio activo.

Sintaxis:

```
cd [directorio]
```

directorio es opcional, si no se especifica, el directorio activo pasará a ser el directorio home (directorio personal) del usuario que ejecuta el comando.

Comando mkdir

Sintaxis:

```
mkdir directorio1 [directorio2 ...]
```

Crea el directorio o directorios que se especifiquen. Es posible crear un árbol completo de directorios con un único comando.

Comando rmdir

Elimina los directorios especificados siempre que estén vacíos. No se puede eliminar el directorio activo.

Sintaxis:

```
rmdir directorio1 [directorio2 ...]
```

Comando ls

Lista el contenido de un directorio.

Sintaxis:

```
ls [opciones] [fichero]
```

Lista información del fichero que se especifique. Si no se especifica ninguna, lista la información del directorio actual.

Si no se pone ninguna opción, lista los archivos sin ninguna información adicional, solo los nombres, en orden alfabético.

Las opciones pueden ser varias:

-a, -all : lista todos los archivos incluidas los ocultos (empiezan por punto) y entrada actual y anterior

-l : lista con el formato largo que incluye también información de permisos de archivos

-r : invierte el orden

-t: ordena por fecha de modificación, los más nuevos antes

Ejemplo:

\$ ls -altr

Lista del directorio, todos los archivos, ocultos y entradas actual y anterior, ordenados por fecha de modificación, los más nuevos al final.

6.-Comandos para la manipulación y gestión de ficheros

Comando cat

Concatena el contenido de los ficheros que se le pasen, si no se le pasa nombres de fichero, lee datos de la entrada estándar (teclado) hasta que encuentre un final de fichero Ctrl+D y entonces produce un eco en la salida estándar (pantalla).

Sintaxis:

```
cat [-n] [fichero1 [fichero2 ...]]
```

-n : El comando produce una numeración de todas las líneas del resultado.

Comando cp

Copia ficheros.

Sintaxis:

```
cp [opciones] origen destino
```

```
cp [opciones] origen1 [origen2 ...] directorio_destino
```

La segunda sintaxis de **cp**, copia varios archivos en un directorio dado.

Los permisos de los archivos también se copian.

origen, origen1, origen2 : son nombres de ficheros (especificados con su ruta relativa o absoluta).

destino: puede ser el nombre de un archivo (con su ruta absoluta o relativa) o un directorio. Si es un directorio, entonces origen es copiado sin cambiar el nombre. Si se quiere copiar en el propio directorio es necesario especificar otro nombre. En cualquier caso si se da un nombre de fichero ya existente su contenido será reemplazado.

Opciones: -r : Copia directorios recursivamente.

Comando mv

Cambia el nombre a un fichero o directorio.

Sintaxis:

```
mv fichero1 fichero2
```

```
mv fichero1 [fichero2] directorio
```

En la primera sintaxis fichero1 es el nombre original de un archivo o directorio y fichero2 es el nuevo nombre.

Comando rm

Elimina archivos o directorios.

Sintaxis:

```
rm [-opciones] fichero1 [fichero2 ...]
```

Opciones:

-r : Permite aceptar nombres de directorios como argumentos, y realiza un borrado recursivo del mismo. Atención a esta opción que puede ser peligrosa.

Si el nombre del fichero (o ficheros) es un directorio obligatoriamente debe utilizarse la opción '-r'.

Comando find

Busca ficheros recursivamente a partir del directorio que se especifique.

Sintaxis:

```
find nodo_ruta criterio_nombre_expresión
```

nodo_ruta : especifica a partir de dónde buscar los archivos (camino).

criterio : indica cómo seleccionar los archivos, si es por el nombre solamente o es por otro criterio cualquiera.

acción : se refiere a, qué hacer con los ficheros seleccionados.

Criterios:

-name fichero : busca ficheros con nombre fichero.

-type f : busca ficheros ordinarios.

-type d : busca directorios.

-user nombre_usuario : busca ficheros cuyo propietario sea el usuario nombre_usuario.

-group nombre_grupo : busca ficheros cuyo grupo propietario sea nombre_grupo.

-atime -n : busca ficheros a los que se ha accedido hace menos de n días.

Ejemplos:

```
find /usr -name fic
```

busca a partir del directorio /usr ficheros llamados fic

```
find / -name fic -type f -user juan
```

busca a partir del directorio raíz los ficheros ordinarios que se llamen fic y tengan como propietario a juan.

Comando ln

Establece un enlace o vínculo entre dos ficheros, de manera que al actualizar uno de ellos los cambios se producirán en los dos.

Sintaxis:

```
ln [-opciones] fichero_existente fichero_nuevo
```

Opciones: -s : crea un enlace simbólico. Si no se especifica el enlace será físico.

Comando grep

Busca cadenas en un fichero o ficheros

Sintaxis:

```
grep [-opciones] cadena fichero1 [fichero2 ...]
```

cadena : cadena de texto a buscar, si contiene espacios en blanco debe ir entre comillas (simples o dobles). Opciones:

-i : no hace distinción entre mayúsculas y minúsculas.

-n : numera las líneas de salida. El número de cada línea será el orden que tiene en el fichero en el cual se encuentra.

-v : Para que el resultado sean las líneas que no contienen la cadena. Si no se especifica esta opción, mostrará las líneas que contienen la cadena.

-c : En lugar de mostrar líneas, muestra el número total de líneas que contienen la cadena en cada fichero.

Comando more

Permite visualizar línea a línea o pantalla a pantalla el contenido del fichero o ficheros que se le pasen si el contenido de estos no cabe en una pantalla.

Sintaxis:

```
more fichero1 [fichero2 ...]
```

Una vez que tengamos llena una pantalla, y siempre que queden más líneas por visualizar, el programa espera la pulsación de alguna tecla. Si pulsamos <ENTER>, la salida avanzará una línea; si pulsamos la barra espaciadora, la salida avanzará una pantalla; por último, si deseamos acabar, podemos hacerlo con <Ctrl>+C.

Comando sort

Da como resultado el contenido de uno o varios ficheros, ordenando las líneas según determinados criterios.

Sintaxis:

```
sort [-opciones] fichero1 [fichero2 ...]
```

Opciones: -r: invierte el orden.

-t separador : especifica el carácter a utilizar como separador de campos. Si no se especifica, se asume que es el carácter espacio en blanco.

-k pos1[,pos2] : Indica los campos por los que ordenar (pos1 y pos2 son valores enteros mayores que cero, 1 para el primer campo, 2 para el segundo campo,...) Si sólo se especifica pos1 se ordenará por el campo que indique, si se especifica pos2, se ordenará en primer lugar por el campo pos1, a continuación (en caso de coincidencia de varios valores en el campo1) por el campo pos1+1 y así hasta pos2. Si no se especifica esta opción, ordenará por el primer campo.

Si no se especifica separador ni campos, la salida se ordenará por líneas. -o fichero_salida : el resultado se guardará en fichero_salida.

Comando tr

Visualiza por la salida estándar el contenido de un fichero sustituyendo unos caracteres por otros.

Sintaxis:

tr [-s] caracter_inicial caracter_final fichero

Devuelve el contenido de fichero, sustituyendo las apariciones del carácter caracter_inicial por caracter_final.

Con la opción `-s` sustituye repeticiones del carácter caracter_inicial por un único carácter caracter_final.

Comando cut

Extrae las partes seleccionadas de cada fichero en la salida estándar. Por defecto usa el tabulador como separador de campo.

Este comando nos permite sacar campos o simplificar ficheros mucho más complejos. Es muy útil para interpretar la salida de un comando y/o fichero y filtrarlo para quedarnos con la información que queremos realmente.

Sintaxis:

cut [-opciones] [fichero]

Opciones:

-d delimitador: Se especifica el «delimitador», el elemento que separa los campos.

-f numero_campo: Especificamos el número del campo. Se puede indicar varios campos, separados por comas (1,2,4 → campos 1, 2 y 4) o separados por guión (1-4 → campos del 1 al 4).

-c numero_caracteres: Especificamos los caracteres que deseamos mostrar.

Ejemplos:

\$ cut -f 11 → selecciona solamente el campo 11

\$ cut -d " " → usa un espacio, en vez del carácter tabulador para delimitar los campos.

\$ cut -d: -f1 /etc/passwd → lista los usuarios del sistema.

\$ cut -d ":" -f1 /etc/passwd → lista los usuarios del sistema.