|  |  |  |
| --- | --- | --- |
| **COMANDO** | **FUNCION** | **EJEMPLO** |
| **man** | **Consulta** el manual del sistema sobre un comando | man <comando>  man ls |
| **ls** | **Listar** archivos y directorios.  Parámetros:   * -l: Información + detallada. * -a: Elementos ocultos. * \*: Recursivo, mostrara contenido de carpetas contenidas en directorio. | ls [opciones] [ruta]  ls -a /etc |
| **cd** | Desplazarnos a un directorio.   * cd. (quedas en carpeta actual) * cd.. (subes al directorio superior/padre). Ruta relativa. | cd [directorio]  cd Documents |
| **mkdir** | **Crea** un directorio vacío. | mkdir [opciones] directorio  Varios: mkdir dir1 dir2 dir3  Recursivo: mkdir -p dir1/dir2/dir3 o  mkdir dir1/dir2/dir3 |
| **rmdir** | **Borrar** un directorio vacío. | rmdir [opciones] directorio  rmdir mi\_directorio  Recursivo: rmdir -p dir1/dir2/dir3 |
| **rm** | **Borra** fichero y directorios.   * Varios ficheros de un directorio: \*. * Directorios y subdirectorios forzando borrado: -rf. | rm [opciones] archivo(s) / directorio(s)  Varios: rm archivo1.txt archivo2.txt archivo3.txt  Eliminar sin confirmación y forzada: rm -rf directorio |
| **touch** | **Crear** archivos vacíos | touch [opciones] archivo  touch nuevo\_archivo.txt  Varios: touch archivo1.txt archivo2.txt archivo3.txt |
| **cat** | **Mostrar** contenido de archivos, **concatenar**, **crear** o **combinar archivos** de texto. | cat [opciones] archivo(s)  cat archivo.txt  Concatenar: cat archivo1.txt archivo2.txt  Crear nuevo: cat > nuevo\_archivo.txt  Agregar contenido: cat >> archivo\_existente.txt |
| **cp** | **Copia** fichero.   * \*: Copia todos los ficheros de un directorio. * -r: Copia subdirectorios de manera recurrente. | cp [opciones] origen destino  cp archivo1.txt archivo2.txt  Copia archivo a directorio: cp archivo.txt /home/usuario/documentos/  Todos archivos: cp \* /home/usuario/documentos/  Recursivo: cp -r directorio1 /home/usuario/documentos/  Recursivo y mostrando los copiados: cp -r -v directorio1 /ruta/destino/ |
| **mv** | **Mover** directorios y ficheros (**elimina** cada **origen** una vez copiado en destino). | mv [opciones] origen destino  Archivo a directorio: mv archivo.txt /ruta/destino/  Renombrar: mv archivo.txt nuevo\_nombre.txt o  mv directorio1 directorio\_renombrado  Varios: mv archivo1.txt archivo2.txt archivo3.txt /ruta/destino/ |
| **>** | **Redirección**, **sobrescribe** el fichero destino.  >>: Si queremos **añadir** contenido al final de un fichero existente. | Redirigir salida a un archivo: echo "Hola, mundo" > archivo.txt |
| **echo** | **Mostrar** cadena de texto que recibe como parámetro.  Introduce un salto de línea – Evitarlos (echo -n “texto”). | echo [opciones] [texto o variables]  echo "Este es un mensaje" > archivo.txt  echo -n "Este texto no tendrá salto de línea al final." |
| **wc** | **Cuenta**   * -l: Líneas. * -w: Palabras. * -c: Caracteres. | wc [opciones] [archivo]  wc -l archivo.txt  wc -w archivo.txt  wc -c archivo.txt  Combinados: wc -lwc archivo.txt o  wc -l -w -c archivo.txt |
| **|** | **Combinar** varios **comandos** para ejecutarlos en un único mandato.  Salida del 1º se manda a entrada del 2º. | comando1 | comando2  Listar archivos y contar número elementos: ls | wc -l  Buscar palabra en archivos de un directorio: ls | grep "archivo"  Ver contenido archivo y buscar palabra: cat archivo.txt | grep "palabra"  Combinación: cat archivo.txt | grep "error" | sort | uniq |
| **ps** | Con “aux” nos muestra todos los procesos que se están ejecutando. | ps [opciones]  ps aux  Buscar un proceso especifico: ps aux | grep "firefox" |
| **kill** | **Finaliza** un proceso en ejecución. | kill [opciones] [PID]  Suave: kill 1234  Forzada: kill -9 1234 |
| **killall** | **Finaliza** todos los programas de un tipo. | killall [opciones] nombre\_del\_proceso  killall Firefox  killall -9 firefox |
| **grep** | **Búsqueda cadena** de caracteres.   * -i: No tiene en cuenta mayúsculas y minúsculas. * -v: Devuelve inverso (líneas que no cumplan patrón). * -n: Numero líneas sucede coincidencia. * -w: Coincidencia solo si el patrón coincide con la palabra completa | grep [opciones] patrón [archivo...]  grep "hello" archivo.txt  grep -i "hello" archivo.txt  grep -v "hello" archivo.txt  grep -n "hello" archivo.txt  grep -w "hello" archivo.txt |
| **find** | **Buscar** ficheros en una ruta determinada. | find [ruta] [criterios] [acción]  Por nombre archivo: find . -name "archivo.txt"  Sin distinguir mayúsculas/minúsculas: find . -iname "archivo.txt"  find /home/usuario -name "documento.txt" |
| **head** | Acompañado de un numero natural “-n” **muestra** las “**n primeras líneas**” del fichero. | head [opciones] [archivo...]  Mostrar primeras 5 líneas: head -n 5 archivo.txt  Mostrar todo menos las ultimas 5: head -n -5 archivo.txt |
| **tail** | Igual a “head” pero **muestra** las **ultimas líneas**. | tail [opciones] [archivo...]  Mostrar primeras 5 ultimas: tail -n 5 archivo.txt |
| **awk** | **Procesa patrones** en líneas de texto.  Acceder a datos con formato tabla.  Acceder a datos separados por un delimitador. | awk 'comando' archivo  Imprimir columnas: awk '{print $1}' archivo.txt  Imprimir líneas primer campo es > 10: awk '$1 > 10 {print $0}' archivo.txt  Operaciones: awk '{print $1 + $2}' archivo.txt |

|  |  |  |
| --- | --- | --- |
| **EXPRESIONES REGULARES** | | |
| **^** | Cadena debe aparecer al comienzo de una línea | grep "^texto" archivo.txt |
| **$** | Cadena debe aparecer al final de una línea | grep "texto$" archivo.txt |
| **[]** | Definir un conjunto de caracteres posibles | grep "[a-z]\*texto" archivo.txt |
| **[0-9]** | Cualquier digito de 0 al 9. | grep "U[0-9]" archivo.txt |
| **\*** | Carácter puede repetirse de 0 a n veces | grep "[a-t]\*texto" archivo.txt |
| **\{1,2\}** | Carácter anterior debe repetirse entre 1 y 2 veces | grep "U[0-1]\{1,2\}" archivo.txt |

|  |  |
| --- | --- |
| **cd** | Cambia directorio |
| **cd .** | Queda en directorio actual |
| **cd ..** | Sube nivel en jerarquía de directorios |
| **cd ../..** | Sube dos niveles en jerarquía de directorios |
| **cd /home** | Cambia al directorio /home |
| **cd ~HOME** | Cambia al directorio del usuario especificado |
| **pwd** | Muestra el directorio actual |
| **ls** | Lista contenido del directorio actual |
| **ls -l** | Llista contenido detallado de directorio |
| **ls -a** | Lista contenido, incluyendo ocultos |
| **ls \*cadena\*** | Muestra archivos que coinciden con cadena |
| **ls h\*** | Muestra archivos comienzan con “h” |
| **ls h???** | Muestra archivos comienza con “h” seguidos de 3 caracteres |
| **mkdir** | Crea un nuevo directorio |
| **rmdir** | Elimina un directorio vacío |
| **rm** | Elimina un archivo |
| **rm \*** | Elimina todos los archivos del directorio actual |
| **rm -rf dir1** | Elimina un directorio y su contenido de forma recursiva |
| **mv** | Mueve o renombra archivos/directorios |
| **cp** | Copia un archivo |
| **cp \* dir\_origen dir\_destino** | Copia todos los archivos |
| **sort** | Ordena líneas de un archivo |
| **sort -r** | Ordena en orden inverso |
| **awk** | Herramienta procesamiento de texto |
| **cat** | Muestra contenido de un archivo |
| **cat file | command** | Canaliza la salida de un archivo a otro comando |
| **echo** | Imprime el texto/variables |
| **echo "texto" > file.txt** | Escribe texto en un archivo (sobrescribe si ya existe) |
| **echo "texto" >> file.txt** | Añade texto a un archivo existente |
| **wc -l** | Cuenta número de líneas de archivo |
| **ps aux** | Muestra información sobre procesos en ejecución |
| **kill** | Mata un proceso por su PID |
| **killall** | Mata todos los procesos por su nombre |
| **find . -name "\*txt"** | Busca archivo con extensión “.txt” en el directorio y subdirectorio |
| **head -n 1** | Muestra las primeras líneas de un archivo (por defecto 10) |
| **tail -n 3** | Muestra las ultimas líneas de un archivo (por defecto 10) |
| **grep [opciones] [exp regular] [fichero]** | Busca un patrón en un archivo |
| **grep -i** | Ignora mayúscula y minúsculas en la búsqueda |
| **grep -v** | Invierte el resultado de la búsqueda (no sea patrón) |
| **grep -n** | Muestra número líneas en que se encuentra el patrón |
| **grep -w** | Busca el patrón como una palabra completa |
| **grep -A n** | Muestra n líneas siguientes a las que contiene el patrón |
| **grep -B n** | Muestra n líneas anteriores a las que contiene el patrón |
| **grep -b** | Muestra desplazamiento de bytes de líneas que contiene patrón |
| **grep '^texto' file.txt** | Busca cadena que empieza con el patrón |
| **grep 'texto$' file.txt** | Busca cadena que termina con el patrón |
| **grep '...to' file.txt** | Busca cualquier carácter |
| **grep '[a-z]texto' file.txt** | Busca conjunto caracteres dentro [] |
| **grep '[a-z]\*' file.txt** | Busca cadena que contenga 0 o + de caracteres dentro [] |
| **grep 'U[01]1' file.txt** | Busca cadena coincida expresión regular |
| **grep 'U[01]1]\{1,2\}' file.txt** | Busca cadena coincida con patrón repetido de 1 a 2 veces |
| **grep 'patron' ./file1 ./file2** | Busca patrón en varios archivos |
| **grep 'patron' ./\*** | Busca patrón en todos los archivos del directorio |
| **grep '.. /lib \*'** | Busca patrón con expresión regular |
| **grep home/usuario/\*.txt** | Busca patrón en todos los archivos del directorio |
| **grep -r** | Busca patrón recursivamente |
| **git init** | Inicializa un repositorio Git |
| **git status** | Muestra el estado del repositorio |
| **git add 'fichero'.txt** | Añade un archivo al área de preparación |
| **git commit -m 'texto descriptivo'** | Hace un commit con mensaje descriptivo |