

TEMA 03 – LINUX. ADMINISTRACIÓN Y CONFIGURACIÓN

Uso del sistema operativo Linux

- Uso del SO Linux
 - o Casesensitive
 - o Entorno de escritorio por defecto de Ubuntu: Gnome (a partir 42)
 - Otros: KE, Xfce, LXDE, Cinnamon
 - o Gestor de ventanas: controla gestión, aspecto y posición de las ventanas en una interfaz gráfica de usuario (GUI)
 - Ofrece aplicaciones, integración entre ellas y una experiencia de usuario más completa
 - Existen: IceWM, Fluxbox, Openbox...
 - o Interfaz gráfica
 - Escritorio: fondo escritorio (carpeta en el fichero personal)
 - Botón Aplicaciones: esquina inferior izquierda
 - Dock: barra lateral izquierda (apps abiertas y favoritas)
 - Menú del sistema (superior derecha): conf, bloqueo, cierre...
 - Selector de ventanas – Actividades: ventanas abiertas
 - Menú aplicaciones: gestionar la aplicación que esté en primer plano
 - Calendario y notificaciones
 - o Terminal
 - Muchas tareas son necesarias realizarlas mediante comandos
 - Atajo: Ctrl+Alt+T o botón secundario y “Abrir en una terminal”
 - Se puede acceder directamente a terminales o consolas virtuales (Ctrl+Alt+(F1-F7))
 - Normalmente la 2 o la 7
 - Comando: ‘sudo chvt n’ (n equivale al número de terminal)
 - #: usuario normal
 - #: usuario root
 - Sintaxis general de los comandos
 - ‘comando [-o | --opción] ... [argumentos] ...’
 - Opción: para cambiar el comportamiento de ese comando o la info que muestra
 - Argumento: indicar sobre qué objeto o dato se va a ejecutar ese comando
 - Comandos iniciales
 - ‘history’: muestra los últimos comandos utilizados
 - -c: borra el historial
 - ‘clear’: borra la pantalla
 - ‘date’: muestra o cambia la fecha
 - ‘date [opciones]... [+formato]’
 - -s, --set=STRING: establece la hora que se le indique en STRING
 - En formato:

%d	Día del mes (por ejemplo, 01)	%H	Hora (00..23)
%m	Mes (01..12)	%I	Hora (01..12)
%y	Los 2 últimos dígitos del año	%M	Minutos (00..59)
%Y	El año	%S	Segundos (00..60)
%D	La fecha; igual que %m/%d/%y	%T	Hora; igual que %H:%M:%S
%r	La hora en formato de 12 horas	%j	Día del año (001..366)
%R	La hora en formato de 24 horas; igual que %H:%M	%F	Fecha completa; igual que %+Y-%m-%d

- Ejemplos:

- ‘date’: fecha en formato completo
- ‘date +%D’: formato dd/mm/yy
- ‘date +%T’: formato hh:mm:ss
- ‘date +%F’: yyyy-mm-dd
- ‘whoami’: muestra el nombre del usuario
- ‘who’: muestra quién está conectado al sistema
 - -a: muestra la fecha de arranque del sistema, los usuarios conectados, la terminal donde están y el nivel de arranque del sistema
- ‘sudo’ (super user do): permite ejecutar otros comandos con privilegios
 - También para ejecutar comando como otro usuario
 - No todos los usuarios pueden utilizar este comando
- ‘uname’: muestra info sobre el SO y el equipo
 - -a: muestra toda la información (menos -p y -i si se desconoce)
 - -s, --kernel-name: nombre del kernel
 - -n, --nodename: nombre de red del equipo
 - -r, --kernel-release: versión de publicación del kernel
 - -v, --kernel-version: versión y fecha de compilación
 - -p, --processor: tipo de procesador o unknown
 - -i, --hardware-platform: tipo hardware o unkwnown
 - -o, --operating-system: muestra el SO
- ‘lsb_release’: info distribución, version Linux y versión LSB (Linux Standard Base, base estándar de Linux)
 - Proyecto que desarrolla un conjunto de estándares para que las distribuciones Linux sean + compatibles entre sí
 - Info en /etc/lsb-release
 - -i, --id: muestra la id del distribuidor
 - -d, --description: descripción de la distribución instalada
 - -r, --release: número de la versión
 - -c, --codename: muestra el nombre de la versión
- ‘lshw’: muestra lista completa del hardware (como sudo)
 - Opciones son para el formato de salida (-html, -xml, -json, -short...)

- ‘passwd’: cambia la contraseña de un usuario
 - ‘passwd [opciones] [usuario]’
 - Sin opciones ni argumentos, cambia la del usuario
 - Para otros usuarios, sudo
 - -d, --delete: la deja vacía
 - -e, --expire: fuerza a que caduque y obligará al usuario a cambiarla en el siguiente inicio de sesión
 - -l, --lock: bloquea cuenta (añade un “!” en el campo contraseña del fichero /etc/shadow)
 - -u, --unlock: desbloquea (elimina el “!”)
- ‘exit’: termina la sesión de un usuario o sale de la terminal
- ‘logout’: sale de la sesión en una terminal virtual o terminal de entrada (no el modo gráfico)
- ‘reboot’: reinicia sistema
- ‘shutdown0: apaga, pausa o reinicia el sistema
 - ‘shutdown [opciones] [tiempo]’
 - -H, --halt: detiene o pausa el sistema
 - -P, --poweroff: apaga el sistema (default)
 - -r, --reboot: reinicia
 - Tiempo: now (ahora) o número (en minutos)

Gestión de archivos y directorios

- Estructura del sistema de archivos

Nombre	Contenido
/bin	Programas ejecutables necesarios para el arranque y funcionamiento del sistema.
/sbin	Programas ejecutables del superusuario necesarios para el arranque y funcionamiento el sistema.
/boot	Archivos necesarios para el arranque del sistema.
/dev	Archivos de los dispositivos del sistema.
/etc	Configuración del sistema.
/home	Contiene los directorios personales de los usuarios normales.
/lib	Bibliotecas necesarias para que se puedan ejecutar los programas ejecutables.
/media	Unidades de almacenamiento montadas, como pendrives, DVD, carpetas compartidas, etcétera.
/mnt	Aquí se montan sistemas de archivos de forma temporal. Se usa de forma similar al anterior.
/opt	Aplicaciones instaladas sin utilizar los repositorios.
/proc	Directorio temporal con archivos que envían información al <i>kernel</i> o núcleo que son los archivos de los procesos que están en ejecución.
/run	Directorio temporal con archivos necesarios para almacenar en la RAM en tiempo de ejecución ciertos datos de procesos en ejecución.
/snap	Directorios con las aplicaciones instaladas con snap .
/root	Directorio personal del usuario <i>root</i> .
/usr	Aplicaciones instaladas por el usuario.
/tmp	Archivos temporales.
/var	Archivos de variables y <i>logs</i> .
/srv	Directarios de los servidores que se instalen en el sistema.
swapfile	Fichero de intercambio para la memoria virtual del sistema.

- Ficheros en Linux
 - Hasta 255 caracteres
 - NO carácter /
 - . para extensiones
 - Cada fichero tiene un nodo índice o inodo
 - Número de inodo: número entero positivo diferente para cada fichero (salvo enlaces duros)
 - Nombre del fichero

- Permisos del fichero
 - Tipo de archivo (regular, directorio, enlace...)
 - Tamaño en bytes
 - Dispositivo donde se encuentra
 - Enlaces que tiene
 - Propietario del fichero: UID del propietario
 - Grupo del archivo: GID del grupo
 - Fecha de creación, acceso y modificación
 - Bloques: tabla con los bloques donde se encuentra el fichero
- Directorios especiales
 - En todos los directorios siempre hay 2 entradas que se añaden al crear un directorio:
 - ..: referencia al mismo directorio
 - ...: referencia al directorio superior
 - Raíz: /
 - Directorio personal: ~
- Ruta absoluta y ruta relativa
 - Ruta absoluta: desde "/"
 - Única
 - Ruta relativa: desde "."
 - Tantas como directorios haya dentro de ella
 - Contaría también "../.." para ir 2 directorios atrás
- Trabajar con archivos y directorios
 - Modo gráfico
 - Ubuntu: Nautilus
 - Por defecto se abre la carpeta personal
 - Opción de ver archivos ocultos
 - Para acceder a raíz: Otras ubicaciones -> En este equipo
 - Terminal
 - 'ls' (list directory): muestra el contenido de los directorios
 - 'ls [opciones] [fichero]
 - -d, --directory: muestra info sobre el directorio en vez de sobre su contenido
 - -a, -all: muestra también ocultos
 - -l: formato largo
 - -i, --inode: muestra el inodo de cada fichero
 - -h, --human-readable: muestra el tamaño en la mayor unidad de medida
 - 'pwd' (print working directory): muestra la ruta absosuta del directorio actual
 - 'mkdir': crea directorio
 - -p, --parents: crea los directorios padres si es necesario
 - 'cd' (change directory): cambio de directorio, sin nada al personal
 - -: va al directorio donde se ha estado anteriormente
 - ...: sube un directorio
 - ~: directorio personal

- ‘rmdir’: elimina directorios que estén vacíos
 - -p, --parents: borra directorios padres si es necesario
- ‘rm’: elimina ficheros y directorios
 - -i: pregunta antes de cada eliminación
 - -R, -r, --recursive: elimina directorios y su contenido de forma recursiva
- ‘cp’: copia ficheros
 - ‘cp [opciones]... origen... destino’
 - -R, -r, --recursive: copia directorios de forma recursiva
- ‘mv’: mueve uno o varios ficheros (al mismo cambia el nombre)
 - ‘mv [opciones]... fuente destino’
 - ‘mv [opciones]... -t directorio fuentes...’
 - -u, --update: mueve solo si no existe o es anterior en la fecha
 - -i, --interactive: pregunta antes de sobrescribir
 - -f, --force: fuerza la sobrescritura
 - -t, --target-directory=DIRECTORIO: mueve todas las fuentes al directorio destino
- ‘touch’: modifica la fecha de un fiche y, si no existe, lo crea en blanco
- ‘stat’: muestra información sobre el fichero que se le pase como argumento
 - ‘stat [opciones]... fichero...’
 - -f, --file-system: muestra info del Sistema de archivos
- ‘du’: muestra el espacio usado en el disco
 - -h, --human-readable: en la unidad de mayor medida
- ‘df’ (display free): muestra el espacio libre en dispositivos almacenamiento y particiones
 - -h, --human-readable
- ‘ln’ (link): crea un enlace o un enlace simbólico al fichero o directorio que se le pase como argumento
 - Enlace (enlace duro) se puede considerar como otro nombre para el mismo archivo
 - Enlace simbólico: fichero que apunta hacia otro que puede ser una carpeta u otro tipo de fichero (como un acceso directo)
 - ‘ln [opciones]--- fichero_o_directorio enlace’
 - -s, --symbolic: hace que el enlace sea simbólico
 - -i: mostrará el mismo inodo en ambos
- ‘whereis’: muestra la ruta de los ficheros fuentes, los ejecutables y las páginas del manual del comando que se le pase como parámetro
- ‘locate’: busca ficheros rápida- por el nombre
 - No busca por el árbol de directorios, sino en una base de datos que se actualiza con el comando ‘updatedb’
 - -b, --basename: busca solo lo que coincide con el nombre del archivo
 - -i, --ignore-case: case insensitive
 - -w, --wholename: busca en la ruta complete del archive (default)

- ‘find’: busca ficheros que coincidan con el patrón por el árbol de directorios
 - ‘find [opciones]... [directorios] [criterios] [acción]
 - -L: sigue los enlaces simbólicos en la búsqueda
 - -maxdepth levels: número de niveles a bajar en la búsqueda
 - Criterios
 - -type {b, c, d, p, f, l, s}: tipo de fichero a buscar
 - Acciones
 - -delete
 - -exec comando \!: para cada fichero que coincide con la búsqueda ejecuta el comando indicado
- Metacaracteres

Carácter	Significado
*	Cero o varios caracteres.
?	Un carácter exactamente.
[]	Un carácter que se indicará dentro del corchete. Se pueden utilizar rangos.
[!]	Cualquier carácter excepto los incluidos dentro del corchete.
{}	Varios caracteres o cadenas de caracteres, separadas por comas.

- Caracteres especiales:

Carácter	Significado
\	Carácter de escape.
'	Comillas simples.
"	Comillas dobles.

- Sustitución de comandos
 - Se puede hacer que la salida de un comando se devuelva en forma de cadena para utilizarla con otro o bien para guardar su valor en una variable
 - \$ (<comando>)
 - ``>comando``
- Separación de comandos y ejecución condicional
 - “;”: separador de instrucciones
 - ||: el segundo se ejecuta si el primero no ha dado un resultado correcto
 - &&: el segundo se ejecuta si el primero se ha ejecutado correcta-
 - Para ejecutar varios comandos en una línea también se pueden emplear los alias

- ‘alias actualiza=”sudo update && sudo upgrade”
- Para ejecutarlo solo se escribe el alias: ‘actualiza’
- Para borrarlo: ‘unalias actualiza’
- Para ver los alias: ‘alias’
- Redireccionamientos y tuberías (pipes)

Tipo	Función
I	La salida estándar (pantalla) de un comando se convierte en la entrada del siguiente comando.
<	Redirecciona la entrada estándar sustituyéndola por el archivo indicado.
>	Redirecciona la salida estándar de un proceso al fichero o dispositivo indicado, borrando la información que el fichero contenía. Si es un fichero que no existe, lo crea.
>>	Redirecciona la salida estándar de un proceso al fichero indicado. No borra el contenido del fichero, sino que añade la información al final del fichero.
2>	Redirecciona la salida de errores de un proceso al fichero indicado, borrando la información que el fichero contenía.
2>>	Redirecciona la salida de errores de un proceso al fichero indicado, pero añadiendo al final de la información del fichero la salida, con lo que no se borra el contenido del fichero.
&>	Redirecciona tanto la salida de errores como la salida estándar a un fichero o dispositivo.
&>>	Redirecciona tanto la salida de errores como la salida estándar a un fichero o dispositivo, pero añadiendo la información al final del contenido del fichero.
2>&1	Redireccionan la salida de errores a donde vaya la salida estándar.

- El 0 indica la entrada estándar (teclado)
- El 1 indica la salida estándar (pantalla)
 - No es necesario ponerlos generalmente
 - El 2 se utiliza para indicar la salida de errores
- ‘cat’ (catenate): concatena ficheros y los muestra en la salida estándar
 - Con un solo fichero como argumento, muestra su contenido
 - -n, --number: muestra el número de cada línea
- ‘head’: muestra la primera parte de un archivo (sin opciones 10 líneas 1as)
 - -c, --bytes=núm: muestra los núm primeros bytes
 - -n, --lines=núm: muestra las núm primeras líneas
- ‘tail’: última parte (mismas opciones que ‘head’ + -f)
 - -f, --follow: muestra las últimas líneas a medida que crece el fichero
- ‘wc’ (word count): muestra el número de líneas, palabras y bytes de un fichero
 - -c, --bytes: muestra nº bytes
 - -l, --lines: muestra nº líneas
 - -w, --words: muestra nº palabras

- ‘more’: muestra el contenido de un fichero y se para a final de pantalla en espera de que el usuario pulse la barra espaciadora o la tecla Intro para ir de línea en línea
 - -n, --number: muestra el número de cada línea
- ‘less’: similar a ‘more’, pero más completo y tiene más funciones
- ‘cut’: elimina secciones de cada línea de uno o varios ficheros
 - -d, --delimiter=delim: utiliza delim como carácter delimitador
 - -f, --fields=list: selecciona los campos a mostrar
 - -c, --characters=list: muestra los caracteres indicados en lista
- ‘sort’: ordena las líneas de un fichero de texto o indica si está ordenado o no
 - -r, --reverse: orden inverso
 - -c, --check: comprueba si la entrada está ordenada, pero no ordena
- ‘grep’: busca coincidencias del patrón en ficheros
 - -e patrón, --regexp=patrón: utiliza patrón para buscar
 - -E, --extended-regexp: utiliza expresiones regulares extendidas (como usar egrep)
 - -i, --ignore-case
 - -v, --invert-match: realiza una búsqueda de lo que no coincide
 - -w, --word-regexp: busca solo palabras completas
 - Las expresiones regulares se pueden hacer con las tablas anteriores y con:

Caracteres	Significado
\d	Dígito.
\w	Palabra.
\s	Espacio en blanco.

Caracteres	Significado
^	Inicio de línea.
\$	Final de línea.
*	Indica que lo anterior se puede repetir cero o más veces.
+	Indica que lo anterior se puede repetir una o más veces.
?	Indica que lo anterior es opcional, es decir, ninguna o una aparición.
{n}	Indica que lo anterior se debe repetir exactamente <i>n</i> veces.
{n,} {,n}	Indica que lo anterior se puede repetir desde <i>n</i> veces o hasta <i>n</i> veces, respectivamente.
{n,m}	Indica que lo anterior se puede repetir desde <i>n</i> hasta <i>m</i> veces.
(a b)	Cualquiera de los dos patrones.
[a-z]	Cualquiera de los caracteres del contenido. Se pueden utilizar rangos.
[^abc]	Negación del contenido.

- Ficheros especiales
 - o Un tipo de archivo especial es el relacionado con las entradas y salidas hacia los dispositivos
 - o Se encuentran en /dev
 - o Se pueden identificar por su nombre y por el 1r carácter de la línea en un 'ls -l'
 - o Ejemplos:

/dev/tty1	- carácter	ls -l /dev/tty*
/dev/pts/0	- carácter	ls -l /dev/pts/0
/dev/sda	- bloque	ls -l /dev/sda*
/dev/null	- carácter	ls -l /dev/null
/dev/cpu	- directorio con dispositivos de carácter	ls -lR /dev/cpu

- o Otros: sockets (s) (permiten la comunicación de procesos en red) y tuberías o pipes (p) (permiten la comunicación entre procesos)

Gestión de usuarios y grupos locales

- Usuarios y grupos
 - o UID (User ID): único de cada usuario
 - o Los usuarios deben pertenecer al menos a un grupo (grupo primario o ppal)
 - o Resto de grupos del usuario: grupos secundarios
 - o Cada grupo tiene un GID (Group ID)
 - o Tipos de usuarios:
 - Usuario root: administrador del sistema o superusuario
 - Tiene todos los privilegios sobre el sistema y acceso a todo
 - Si se puede iniciar sesión con él hay que hacerlo con cautela
 - Su UID es 0 y su directorio personal es /root
 - Usuarios del sistema o especiales: no son usuarios que van a iniciar sesión en el sistema
 - Tampoco son usuarios físicos
 - Suelen ser usuarios que necesitan ejecutar ciertos procesos y servicios a su nombre y ser propietarios de archivos
 - UID 1-999
 - Usuarios estándar o normales: usuarios que iniciarán sesión en el sistema
 - UID de 100 en adelante
 - Directorio personal: /home/carpeta_con_su_nombre
 - o Gestión en el entorno gráfico
 - Se puede agregar usuarios
 - Tras “Desbloquear...”: modificar contraseña, cambiar nombre usuario
 - Admins pueden añadir usuario
 - Tipo de cuenta
 - Nombre completo (nombre usuario)
 - Nombre de usuario (cuando inicie sesión y directorio home)
 - Contraseña
 - Para realizar la gestión de forma más completa, instalar “Usuarios y grupos” (sudo apt install gnome-system-tools)
 - Añadir y eliminar usuarios
 - Gestionar grupos (añadir, eliminar, modificar)
 - Ajustes avanzados: modificar info de contacto, privilegios
 - o En avanzado, desactivar cuenta, cambiar directorio home, Shell, grupo principal y UID
 - o Desactivar usuario modifica el /etc/shadow añadiendo un “!” al inicio del campo
 - o Comandos relacionados con la gestión de usuarios y grupos
 - Mayoría desde ‘sudo’
 - ‘adduser’: añade usuario
 - También para añadir un usuario a un grupo existente
 - --system: crea un usuario del sistema
 - --home DIR: crea el directorio personal donde se indique en DIR

- --shell SHELL: la Shell del usuario predeterminada será la indicada en SHELL
 - --no-create-home: no crea el directorio personal
 - --uid ID: el UID será el indicado en ID
 - --ingroup GROUP | --gid ID: especifica el grupo primario del usuario, que se puede indicar mediante el nombre del grupo en GROUP o mediante su GID en ID
- ‘addgroup’: añade un grupo al sistema
 - --system: añade un grupo del sistema
 - --gid ID: el GID del grupo será el indicado en ID
- ‘deluser’: elimina un usuario del sistema
 - --remove-home: elimina el directorio del usuario eliminado
 - --remove-all-files: elimina todos los ficheros del sistema que pertenezcan a ese usuario
- ‘delgroup’: elimina un grupo de usuarios
 - --only-if-empty (si no, avisa de que no y no borra)
- ‘usermod’: modifica la cuenta de un usuario
 - -d, --home: cambia el directorio personal del usuario
 - -m, --move-home: mueve el contenido del directorio personal al nuevo
 - -a, --append: añade el usuario a los grupos que se indique con -G
 - -G, --groups GRUPOS: lista de grupos separadas por comas
 - -l, --login X: cambia el nombre del usuario
 - -s, --shell X: cambia la Shell de inicio del usuario
 - -L, --lock X: bloquea y “!” en el shadow
 - -U, --unlock X
- ‘groupmod’: modifica un grupo
 - -g, --gid GID: modifica el GID del grupo
 - -n, --new-name: modifica el nombre del grupo
- ‘chsh’: cambia el Shell de inicio del usuario
 - -s Shell: cambia el Shell por el indicado
 - ‘chsh -s /bin/sh usuario’
- ‘id’: muestra info sobre el usuario con su UID y los grupos con su GID
 - Con -un ofrece la misma info que el comando ‘whoami’
 - -g, --group: muestra solo el grupo principal
 - -G, --groups: muestra los GID de los grupos a los que pertenece
 - -u, --user: muestra el UDI
 - -n, --name: muestra el nombre en vez de los números
- ‘groups’: muestra los grupos a los que pertenece el usuario
 - Si no se especifica usuario, muestra la info sobre el usuario que ejecuta el comando
- ‘chown’: cambia el propietario y el grupo de uno o varios archivos
 - -R, --recursive: cambia los subdirectorios de forma recursiva
 - ‘chown usuario fichero.txt’: cambia el propietario a fichero.txt
 - ‘chown usuario:grupo fichero.txt’: cambia propietario y grupo
- ‘chgrp’: cambia el grupo de uno o varios archivos

- -R, --recursive: cambia los subdirectorios de forma recursiva
- 'chgrp --recursive grupo ~/trabajos': cambia el grupo de la carpeta trabajos y todas las subcarpetas
- Ficheros de configuración
 - Se encuentran repartidos en varios lugares de la estructura
 - Contienen info necesaria para la configuración del SO
 - A través de ellos se puede modificar el comportamiento del sistema
 - '/etc/login.defs': contiene los valores predeterminados que tendrá el usuario nuevo
 - '/etc/passwd': info sobre los usuarios del sistema

login:x:UID:GID:información:directorio_personal:shell_de_inicio						
1	2	3	4	5	6	7

- 1: nombre del usuario
- 2: x que hay contraseña encriptada en '/etc/shadow'
- 3, 4: identificadores del usuario y del grupo principal del usuario
- 5: info sobre el usuario (campos GECOS)
- 6: directorio personal
- 7: Shell que se ejecutará por defecto
- '/etc/shadow': contiene info sobre la contraseña de cada usuario y de la contraseña cifrada
 - Accesible a root por seguridad
- '/etc/group': info sobre los grupos definidos en el sistema
 - Usuarios pertenecen a grupo primario o principal (suyo)
 - Opcional- a otros
 - La "x" de la contraseña hace referencia al '/etc/gshadow'
 - Nombre del grupo, x, GID, lista por comas de integrantes
 - 'grupo:x:GID:usuarios'
- '/etc/gshadow': contraseñas de los grupos
 - Acceso solo root
 - 'nombre:contraseña:administradores:miembros'
- '/etc/deluser.conf': configuración con los parámetros predeterminados cuando se utilizan los comandos 'deluser' y 'delgroup'
- '/etc/adduser.conf': configuración con los parámetros predeterminados cuando se usan los comandos 'adduser' y 'addgroup'
- '/etc/shells': ruta de las shells o intérpretes de comandos válidos
 - Si tiene '/bin/false' o '/usr/sbin/nologin' no podrá iniciar sesión
- '/etc/skel': directorio que tiene el contenido del directorio de los usuarios que se añadan al sistema
- Cambiar de usuario o ejecutar comandos con privilegios de otro usuario
 - 'su': cambia de usuario
 - Si no se indica nada, como root
 - Pide propia contraseña
 - -, -l, --login: ejecuta los scripts de inicio de sesión del usuario

- ‘sudo’: permite ejecutar comandos con permisos o privilegios root
 - También por otro usuario si se especifica
 - Previamente pertenecer a grupo sudo o estar en ‘/etc/sudoers’
 - Solicitará la clave propia
 - -i, --login: ejecuta los scripts de inicio de sesión del usuario (parecido a ‘su -l’)
 - -l, --list: si no se indica comando, muestra qué tiene permitido hacer el usuario
 - -u <usuario>, --user=<usuario>: realiza la acción como el usuario indicado en vez de como superusuario o root
 - -e, --edit: edita un fichero con privilegios de root o del usuario específico
- ‘visudo’: editor para el contenido del fichero ‘/etc/sudoers’
 - La 1^a vez preguntará qué editor usar
- Directorio personal del usuario e inicio de sesión
 - ‘/home_nombre_usuario’
 - El del usuario root: ‘/root’
 - ‘sudo su’, ‘sudo -i’: para entrar como root desde otro usuario
 - No hace falta conocer la contraseña de root, sino la propia
 - Hay que pertenecer al grupo sudo

Permisos

- Todos los archivos pertenecen a un usuario y a un grupo
- Por defecto, usuario que crea y grupo primario
- 3 grupos de permisos
 - o Propietario del grupo (u)
 - o Pertenecientes al grupo (g)
 - o Resto (o)
- 'chmod': establece los permisos de un archivo o directorio
 - o 'chmod [opciones] permisos[, permisos]... archivos...
 - o 'chmod [opciones] numero_octal archivos...
 - o Modo simbólico:
 - 'chmod [ugoa] [[-+=] [rwxXst]] archivos...
 - o Tabla permisos:

Permiso	Significado
r	Permiso de lectura.
w	Permiso de escritura.
x	Permiso de ejecución en ficheros; buscar o poder entrar en directorios.
X	Permiso X mayúscula: añade permiso x a los directorios y en el caso de que algún fichero contenga algún permiso de ejecución se le amplía ese permiso de ejecución para todos los usuarios.
s	Permiso setuid bit o setgid bit , que indica que al ejecutar un fichero el proceso pertenecerá al propietario del fichero o al grupo, respectivamente, no al usuario que ejecutó el fichero.
t	Permiso sticky bit (bit pegajoso), que se aplica sobre directorios y permite que un usuario si tiene permisos de escritura pueda modificar los archivos, pero le impide eliminarlos del directorio.

- o Modo numérico:
 - (Especiales, si los hay,) Usuario, grupo, resto
 - 0: ningún permiso
 - 1: ejecución
 - 2: escritura
 - 4: lectura

Permisos	Valor binario	Valor numéricico
---	000	0
- -x	001	1
-w-	010	2
-wx	011	3
r--	100	4
r-x	101	5
r w-	110	6
rwx	111	7

Gestión de procesos y servicios

- Introducción
 - o Proceso/tarea: programa que está en ejecución
 - o Cada vez que se ejecuta un programa se crea uno o varios procesos
 - o Puede haber más de un proceso o instancia de un mismo programa
 - Propiedades, contexto... (usuario que lo lanza, dueño del proceso, permisos del proceso, si ha sido llamado por otro proceso...)
 - o Cada proceso se identifica por un PID (Process ID)
- Gestión de procesos
 - o Monitor del sistema
 - En la utilidad “Monitor del sistema”, pestaña “Procesos”
 - Se ven procesos en ejecución e info
 - Puede estar en segundo plano durmiendo o en ejecución
 - Botón derecho en el proceso: detener, continuar, finalizar, matar, ver propiedades, cambiar prioridad
 - o Comandos para gestionar procesos
 - o ‘ps’: informa del estado de los procesos que se están ejecutando en el sistema
 - -A, -e: muestra todos los procesos
 - -l: formato largo
 - -y: muestra la cantidad de memoria física
 - ‘a’: muestra todos los procesos con su terminal
 - ‘u’: utiliza un formato adecuado para ser visualizado
 - ‘x’: muestra todos los procesos aunque no tengan asignada una terminal
 - o La información que muestra ‘ps’ es en un instante determinado, para que se vaya refrescando:
 - ‘top’ o ‘htop’
 - Si ‘htop’ no viene instalado por defecto: ‘sudo apt install htop’
 - o ‘pstree’: muestra un árbol con los procesos del sistema
 - o ‘nohup’
 - Estos comandos se utilizan para poder ejecutar procesos en segundo plano y así dejar libre la terminal
 - Se utilizan para procesos que no necesitan interacción con el usuario
 - Se pueden utilizar de alguna de las siguientes formas:
 - ‘nohup <comando>’
 - ‘<comando> &’
 - o ‘nice’: ejecuta los procesos con una prioridad más baja o muestra la prioridad determinada
 - ‘nice [-n <prioridad>] [comando]’
 - Prioridad: -19 (mayor prioridad) a 20 (menor prioridad)

- ‘renice’: modifica la prioridad de un proceso en ejecución
 - ‘renice [opciones] <identificador>’
 - ‘-p, --pid’: interpreta qué identificador es el PID del proceso
 - ‘-n, --priority’: *identificador* es la prioridad nueva del proceso
 - ‘-u, --user’: *identificador* son los nombres o UID de los usuarios
- ‘jobs’: muestra los trabajos activos asociados a una terminal
 - ‘-l’: muestra demás el PID de cada proceso
- ‘fg’: envía a ejecutarse a primer plano un trabajo que estuviese en segundo plano
 - ‘fg [id_trabajo]’
 - Sin operando selecciona el primer trabajo mostrado con el comando ‘jobs’
 - Si se quiere seleccionar otro, se escribirá ‘%2’ para indicar el segundo, ‘%3’ para el tercero y así sucesivamente
- ‘bg’: envía a ejecutarse en segundo plano un trabajo que estuviese parado
 - ‘bg [id_trabajo]’
 - Ídem funcionamiento a ‘fg’
- ‘kill’: por defecto mata el proceso
 - Si se van a matar varios, se escribe el PID de todos ellos separados por un espacio
 - Se puede escribir el nombre del proceso en vez del PID
 - ‘killall’: mata todos
 - ‘-u, --user <usuario>’: procesos asociados al nombre de usuario
 - ‘-i, --interactive’: pide confirmación antes de terminar cada proceso
- ‘time’: ejecuta un programa y muestra el tiempo de uso de los recursos del sistema
- Gestión de servicios
 - Introducción
 - Proceso que se está ejecutando en segundo plano
 - Está a la espera de ser llamado por el usuario u otro proceso para realizar la tarea para la que ha sido programado
 - En Linux: daemons (Disk and Execution Monitor), demonio en español
 - ‘systemd’: sistema para gestionar y administrar los procesos y servicios
 - Se puede gestionar a través del comando ‘systemctl’
 - ‘systemctl’
 - Principal utilidad para gestionar lo que se denomina unidades en ‘systemd’
 - Las unidades pueden ser servicios (service), dispositivos (device) objetivos de inicio (target), temporizadores (timer), etc.
 - ‘systemctl [opciones...] comandos...’
 - ‘-t, --type=:’ lista separada por comas de las unidades que listar
 - ‘-a, --all’: muestra todas las unidades
 - ‘sudo systemctl start|stop|restart|reload <nOMBRE_servicio>’
 - ‘sudo systemctl enable|disable<nOMBRE_servicio>’
 - El ‘enable’ es que se iniciará, para que sea ahora ‘—now’
 - El nombre del servicio puede escribirse sin el ‘.service’
 - ‘systemctl -t service list-units –all’: mostrar todos los servicios

Copias de seguridad

- Introducción
 - o Hacerlas de forma planificada y con regularidad
 - o Total o íntegra: realiza una copia de todos los archivos que se quieran respaldar
 - o Incremental: solo se copian los archivos modificados desde la última vez
 - Para restaurarla, hay que recurrir a la última total más las incrementales que se hayan ido realizando
 - o Diferencial: respalda los archivos que hayan sido modificados desde la última copia de seguridad total o íntegra
 - Se necesitaría la última total y diferencial
 - o Conveniente: /home, algunos de /etc y /var/log para poder investigar y descubrir qué causó un posible fallo en el sistema
- Utilidad para copias de respaldo
 - o Ubuntu: Copias de respaldo de Déjà Dup
- Comandos para copias de seguridad
 - o ‘tar’: utilidad para empaquetar archivos y directorios
 - -c, --create: crea un nuevo archivo
 - -f, --file: indica el archivo que será el nombre del paquete de tipo ‘jar’
 - -t, --list: muestra la lista de los ficheros que contiene el paquete
 - -v, --verbose: muestra el proceso de archivo de los ficheros
 - -x, --extract: extrae ficheros del archivo de tipo paquete ‘tar’ que se indique
 - Para comprimir varios ficheros y empaquetarlos en un solo archivo habría que combinar la utilidad ‘tar’ con ‘gzip’, ‘bzip2’ u otro
 - -z, --gzip: comprime el fichero ‘tar’ con la utilidad ‘gzip’
 - El paquete tendría la extensión ‘.jar.gz’ o ‘.tgz’
 - -j, --bzip2: comprime el fichero ‘tar’ con la utilidad ‘bzip2’
 - El paquete tendría la extensión ‘.tar.bz2’ o ‘.tbz2’
 - o Los comandos de compresión también se pueden utilizar directamente después de haber creado el paquete
 - o ‘gzip’ y ‘gunzip’: comprime o descomprime ficheros
 - Reemplaza los ficheros a comprimir por otro del mismo nombre y extensión ‘.gz.’ o ‘.tgz’ (si es un archivo de paquete)
 - -l, --list: muestra información de cada fichero que comprime
 - -r, --recursive: comprime los directorios y su contenido de forma recursiva
 - -f, --force: fuerza la (des)compresión, incluso si el archivo ya existe
 - o ‘bzip2’ y ‘bunzip2’ igual, pero con ‘.bz2’ o ‘.tbz2’

Programación de tareas

- Programar una tarea
 - o 'at': ejecuta la tarea en el instante indicado
 - Puede ser que no esté instalado: 'sudo apt install at'
 - o 'atq': muestra los trabajos pendientes de ejecución
 - o 'atrm': elimina un trabajo por su número
- Ejecutar comandos programados
 - o 'crontab': gestiona los ficheros crontab de cada usuario para ejecutar tareas de manera periódica
 - 'crontab [-u usuario] [opciones]'
 - 'crontab [-u usuario] fichero'
 - -u: usuario al que se le va a administrar el fichero
 - -e: edita el fichero crontab; si no existe, lo crea
 - -l: muestra el contenido del fichero
 - -r: elimina el fichero crontab



- En los cinco primeros campos, además de números, se puede:
 - *: todos los valores
 - ,: se pueden poner valores separados por comas
 - -: rango de valores
 - /: salto en los valores
- También se pueden usar ciertos nombres con el significado siguiente:
 - @yearly: cada año
 - @monthly: cada mes
 - @weekly: cada semana
 - @daily: cada día
 - @hourly: cada hora

Monitorización del sistema

- Rendimiento
 - o Monitor del sistema
 - Gráficamente: Aplicaciones -> Utilidades -> Monitor del sistema
 - Pestañas: Procesos, Recursos y Sistemas de archivos
 - Terminal: 'gnome-system-monitor'
 - o Carga promedio
 - Medida de procesos que compiten por la CPU o procesador en un instante de tiempo determinado
 - En Linux por defecto son 1, 5 y 10'
 - Se puede consultar con estos comandos:
 - 'top' y 'htop': muestran información de los procesos en tiempo real
 - 'uptime': muestra información sobre el tiempo que ha estado funcionando, el número de usuarios y la carga promedio
 - 'w': muestra lo mismo que 'uptime' y muestra los usuarios conectados y en qué terminal
 - Un 0 indica que no hay carga, que sería el mínimo
 - El valor ideal sería <1
 - Significa que la CPU está atendiendo a todos los procesos
 - Valores mayores: que puede haber colapso
- Eventos del sistema
 - o Todo lo que sucede en el sistema se va almacenando en ficheros llamados ficheros de log
 - o La info que contienen se puede ver directa- o utilizar alguna utilidad para verla
 - o Ficheros de 'log'
 - Cualquier actividad que sucede se puede consultar
 - A través de la aplicación Registros
 - 'journalctl': permite acceder a la info contenida en los archivos de evento del sistema o ficheros de log
 - 'jounalctl [opciones] [coincidencias]'
 - 'journalctl -r': muestra todos los eventos en orden inverso
 - 'journalctl -f': similar a 'tail -f /var/log/syslog'
 - 'journalctl -n10': muestra las 10 últimas líneas
 - 'journalctl _UID=1000': muestra los mensajes del usuario cuyo ID se indique
 - 'journalctl -u <servicio>': para controlar los eventos de un determinado servicio

Variables

- Lugar en la memoria donde el sistema o un usuario guarda cierta info que después será necesario recuperar
- Pueden ser locales, de Shell o de entorno
- También están las propias de usuario que puede utilizar en sus scripts
- El sistema usa variables para almacenar valores relativos a su config o necesarios para que este funcione de forma correcta
- Las variables de entorno se utilizan por los procesos ejecutados en un servidor y sirven para los clientes que accedan a este
 - o Las variables locales y las de Shell solo afectan a la sesión activa en cada equipo
- 'printenv': para ver los valores de las variables de entorno se puede usar el comando
- Para ver el valor de alguna variable:
 - o 'printenv <variable>'
 - o 'echo \$VARIABLE'
- 'unset <variable>': elimina variable
 - o También 'env -u, --unset=NOMBRE'
 - Sin opciones, muestra las variables del entorno
- Export <variable>: para que una variable local pase a ser variable de entorno
- Variables más usadas:

Variable	Contenido
HOME	Ruta hacia el directorio personal del usuario.
SHELL	Ruta al intérprete de órdenes.
PWD	Ruta del directorio de trabajo actual.
LOGNAME	Nombre del usuario que está conectado en ese momento.
LANG	Configuración del idioma y localización del sistema.

Variable	Contenido
USER	Nombre del usuario que inició la sesión.
TERM	Tipo de emulación de terminal (puede ser de tipo Linux o de tipo color).
OLDPWD	Ruta del directorio anterior al que hemos estado (si se usa el comando <code>cd</code> con la opción siguiente, <code>cd -</code> , se irá al directorio indicado en la variable).
PATH	Rutas (separadas por el carácter :) donde el intérprete de órdenes buscará los comandos que se vayan a ejecutar, si no se especifica una ruta hacia él.

- 'set': para ver otras variables, como las variables de Shell
 - o Variables de Shell más importantes:

Variable	Contenido
HISTFILE	Nombre del fichero que guardará el historial de los comandos.
HISTFILESIZE	Número de comandos que se pueden guardar en el fichero .bash_history .
HISTSIZE	Número de comandos que están en la caché.
HOSTNAME	Nombre del equipo.
PS1	<i>Prompt</i> primario del sistema.
PS2	<i>Prompt</i> secundario del sistema.

- Las variables se cargan en los ficheros de inicio y fin de sesión del sistema y de cada usuario
- El usuario puede crear y utilizar sus propias variables:

variable=valor	Ejemplo: num=7; echo \$num
variable=`comando`	Ejemplo: hoy=`date +%D`; echo \$hoy
variable=\$(comando)	Ejemplo: hora=\$(date +%H:%M:%S); echo \$hora

Introducción a los scripts de Linux

- Introducción
 - o Serie de comandos que se ejecutarán cada vez que se ejecute el script
 - o Para automatizar tareas del SO
 - o Se suelen utilizar para automatizar tareas de administración del sistema, aunque se pueden utilizar para cualquier utilidad
 - o Scripts del sistema que se ejecutan cada vez que el usuario inicia o cierra sesión:
 - '.bash_profile'
 - '.bash_logout'
 - '.profile'
 - o Al cerrar sesión: 'bash:logout'
 - o Extensión: .sh (no es obligatoria)
 - o Necesitan permiso de ejecución ('chmod +x')
 - o Incluirlos dentro de la ruta d la variable PATH o indicar la ruta
 - o Primera línea: '#!/bin/bash'
- Comentarios: '# comentario de línea'
- Variables específicas de los scripts

Variable	Función
\$0	Nombre del <i>script</i> .
\$1, \$2, \$3...	Parámetros posicionales que se introducen en el <i>script</i> desde la línea de comandos después del nombre.

Variable	Función
\$#	Número de parámetros posicionales que tiene el <i>script</i> .
\$*	Variable que recoge el valor de todos los parámetros posicionales.
\$?	Valor devuelto por el último comando ejecutado. No solo se utiliza en <i>scripts</i> , sino en cualquier comando que se ejecute en el sistema.
\$\$	PID (<i>Process ID</i> , identificador de proceso) del <i>script</i> . Al ejecutarse, como cualquier otro programa, crea un proceso que tiene un número que lo identifica en el sistema.

- Comandos
 - o 'echo': muestra una línea de texto
 - -n ncars: lee solo los caracteres indicados en ncars
 - -p prompt: muestra la cadena prompt por la pantalla
 - -s: no muestra la salida que se lea
 - -t tiempo: espera un tiempo y expira al cabo del tiempo indicado
 - o 'shift': desplaza a la izquierda los parámetros posicionales
 - El primer parámetro por la izquierda se pierde
- Estructuras condicionales y de control

- ‘if/else’: ejecuta un comando o varios dependiendo de una condición

```

if <condición>
then
    <comando/s>
[elif <condición>
then
    <comando/s>]
[else
    <comando/s>]
fi

```

- ‘for’: ejecuta uno o varios comandos el número de veces indicado en “valores”

```

for $variable in valores
do
    <comando/s>
done

```

- ‘while’: repite un código mientras no se cumpla una condición

```

while <condición>
do
    <comando/s>
done

```

- ‘until’: repite un código hasta que no se cumpla una condición

```

until <condición>
do
    <comando/s>
done

```

- ‘case’: ejecuta un código u otro dependiendo de un valor

```

case $variable in
    expr1) comando/s;;
    expr2) comando/s;;
    ...
    *) comando/s;;
esac

```

- ‘select’: el valor de la variable puede ser uno de los que estén en la lista de valores

- Una vez seleccionado, se ejecuta el comando/comandos que se indique

```

select $variable in valores
do
    <comando/s>
done

```

- Condiciones

Condición	True o verdadero si:
-f valor	<i>valor</i> es un fichero que existe y es un fichero regular.
-d valor	<i>valor</i> existe y es un directorio.
-e valor	<i>valor</i> existe.
-w valor	<i>valor</i> tiene permiso de escritura.
-r valor	<i>valor</i> tiene permiso de lectura.
-x valor	<i>valor</i> tiene permiso de ejecución.
-L valor	<i>valor</i> es un enlace simbólico.
valor1 -nt valor2	<i>valor1</i> es un fichero más reciente que <i>valor2</i> .
valor1 -ot valor2	<i>valor1</i> es un fichero más antiguo que <i>valor2</i> .

Tabla 3.12. Condiciones que se pueden utilizar para comprobar los valores de las cadenas de caracteres

Condición	True o verdadero si:
valor1 = valor2	<i>valor1</i> es una cadena igual que <i>valor2</i> .
valor1 != valor2	<i>valor1</i> es una cadena distinta a <i>valor2</i> .
-n valor	<i>valor</i> es una cadena de longitud mayor que 0.
-z valor	<i>valor</i> es una cadena vacía o de longitud igual a 0.

Tabla 3.13. Condiciones que se pueden utilizar para comparar valores numéricos

Condición	True o verdadero si:
num1 -eq num2	<i>num1</i> es un número igual a <i>num2</i> .
num1 -ne num2	<i>num1</i> es un número distinto a <i>num2</i> .

Condición	True o verdadero si:
<code>num1 -lt num2</code>	<code>num1</code> es un número menor que <code>num2</code> .
<code>num1 -gt num2</code>	<code>num1</code> es un número mayor que <code>num2</code> .
<code>num1 -le num2</code>	<code>num1</code> es un número menor o igual a <code>num2</code> .
<code>num1 -ge num2</code>	<code>num1</code> es un número mayor o igual que <code>num2</code> .

-a o %%: NAD

-o o ||: OR

!: NOT

- Funciones
 - Se pueden declarar funciones para poder llamarlas desde el código

```
function nombre {
    <comando/s>
}
```