Tipos de usuarios

- **Root:** También llamado super usuario o Administrador. Se trata de la cuenta con los privilegios más altos en el sistema operativo. Puede llevar a cabo cualquier función de administración.
- **Usuarios normales:** Tienen pocos permisos, y son sobre los que se aplican todos los que podemos crear para gestionar los accesos Cada usuario tendrá su directorio personal dentro de la /home.
- **Usuarios del sistema :** Incorporados por el propio sistema. Usuarios que no pueden realizar un login, simplemente se dedican a lanzar procesos, entre otros.

ID. DE USUARIO, GRUPO

Identificador de **usuarios** = **"UID"**

- **Usuario ROOT** . . . . : UID es 0.
- **Usuarios normales** : El UID es superior a 1000.
- **Usuarios sistema.** . : El UID es entre 1 y 100.

Identificador de **grupos** = **"GID" o "GROUP ID"**

ID. DE ARCHIVO, CARPETA O ENLACE

Cada archivo o carpeta que nos encontremos al listar el contenido de un directorio irá indicado con un símbolo:

- **–** indica que se trata de un **archivo**.
- **d** indica que se trata de un **directorio**.
- **l** indica que se trata de un **enlace**.

TIPOS DE USUARIOS , PERMISOS y NIVELES EN LINUX

Tipos de permisos

- **(r) lectura:** Es el primer permiso que podemos encontrarnos. Este nos da la opción de que un usuario pueda ver el contenido al que quiere acceder.
- **(w) escritura:** Nos da la posibilidad de otorgar poder sobre sobre un archivo. De esta forma podrá ser modificado, al igual que un directorio.
- **(x) ejecución:** Permite a los usuarios ejecutar diferentes parámetros dentro del equipo.
- **(-) sin permisos:** Nos indica que el usuario no tiene ningún tipo de permiso sobre el recurso de red o contenido compartido.

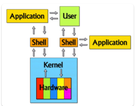
Niveles de permisos

- **(U) Permisos de Propietario:** Usuario que crea el archivo desde su equipo. Linux asigna a este usuario el acceso a la información creada, y la posibilidad de realizar cambios sobre el mismo.
- **(G) Permisos de Grupo:** Cuando un usuario pertenece a un grupo dentro del directorio de Linux, quiere decir que se le otorgan los mismos permisos que tienen los demás usuarios que pertenecen a ese mismo grupo.
- **(O) Permisos del Resto de usuarios:** Usuarios que no son los creadores del archivo, ni pertenecen al grupo que tenga indicado anteriormente. Los permisos y accesos de estos usuarios los puede establecer el propietario.

Permisos		
N°	BINARIO	PERMISOS
0	000	---
1	001	--x
2	010	-w-
3	011	-wx
4	100	r--
5	101	r-x
6	110	rw-
7	111	rwx

(*) Ejemplo de permisos de un archivo en nuestro disco de UNIX

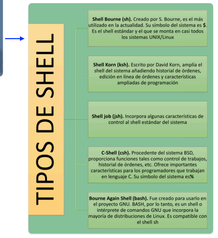
Tipo	Propietario	Grupo	Otros
-	rwx	r--	---
Archivo	lectura-escritura-ejecución	lectura	sin permiso

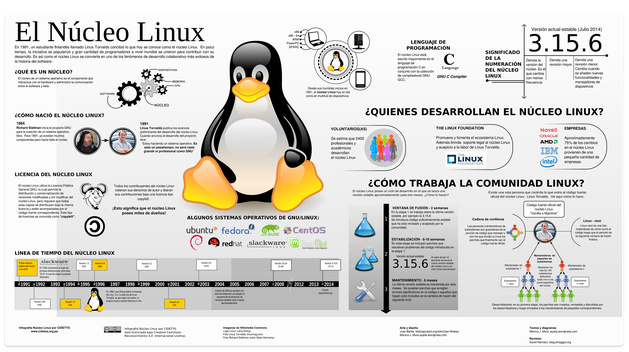


¿Qué es una Shell?

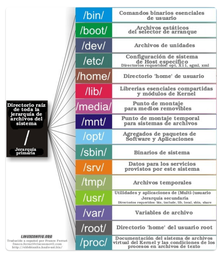
La shell es la interfaz de línea de comandos con la que interactúas. Los ejemplos de Shell son BASH, CSH, ZSH, etc. Estas toman comandos escritos a mano y le dice al sistema operativo que debe ejecutarlos, o puede tomar secuencias de comandos e interpretar la lógica dentro de ellos, además da instrucciones al sistema operativo para ejecutar los comandos contenidos en el script, etc.

El **shell** es el programa que maximiza procesa comandos y devuelve resultados. La mayoría de los shells también gestionan procesos en primer plano y en segundo plano, historial de comandos y edición de línea de comandos. Estas características ya muchas más son estándar en bash, el shell más común en los sistemas GNU/Linux modernos. Aunque existen distintos tipos de shells.

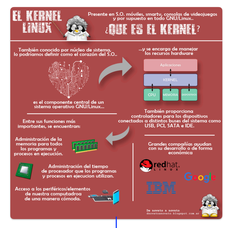
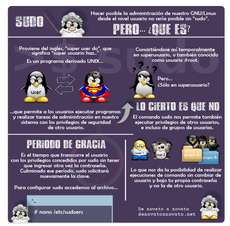
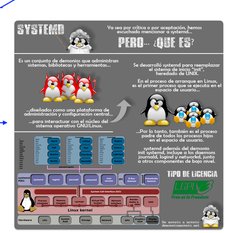




ESTRUCTURA DE CARPETAS EN LINUX



Los 15 comandos Linux básicos

1. **whoami** nos indica el usuario que tiene la sesión de la computadora activa.
2. **pwd** indicará la ubicación en donde se encuentre el cursor.
3. **ls** mostrará una lista de archivos y carpetas que se encuentran en la ubicación actual.
4. **clear** limpiara la terminal del texto que se vaya juntando.
5. **history** nos indica el historial de comandos. Linux que se han ejecutado desde la primera vez que se instaló el S.O.
6. **mkdir** creará una carpeta.
7. **rmdir** borra directorios vacíos.
8. **sudo** asignará los privilegios del administrador. Se utiliza cuando queremos ejecutar un comando con los máximos permisos o para crear archivos que sólo el administrador pueda modificar. También es útil en la actualización de S.O. Por ejemplo:
   1. sudo mkdir holaMundo . Creará a esta carpeta cuyo «dueño» será el administrador.
   2. sudo apt-get update . Ejecuta el programa apt-get a la indica la instrucción «update» para que actualice a las bibliotecas de sus repositorios. Si no se utiliza sudo, no se podrá ejecutar ese programa, ya que este explicita mente requiere tener dichos permisos.
   3. Importante. No es recomendable agregar este comando a alguna instrucción o programa a menos que sea absolutamente necesario.
9. **cd** permite cambiar la ubicación de la terminal para ingresar a una carpeta distinta de la actual.
10. **rm** borrá archivos, por ejemplo.
11. **locate** permite buscar archivos de forma sencilla, aunque no tiene tantas opciones de búsqueda y en algunas ocasiones no encuentra algunos archivos. Por ejemplo, se usa de la siguiente forma:
    1. locate hola.txt y le desplegará en texto las ubicaciones de todos los archivos que terminan en hola.txt.
    Como por ejemplo README-Text-Dyne-Schola.txt
12. **cp** puede copiar un archivo.
13. **chmod** puede cambiar los permisos de un archivo o una carpeta.
14. **touch** permite crear archivos vacíos.
15. **mv** puede mover archivos de una ubicación a otra.

- **pwd** → Directorio actual en el que te encuentras
- **cd** → Cambiar de directorio / carpeta
- **ls** → Ver el contenido de un directorio
- **cat** → Ver el contenido de un archivo
- **cp** → Copiar un archivo
- **mv** → Mover directorios y/o archivos
- **mkdir** → Crear un nuevo directorio / carpeta
- **rmdir** → Eliminar directorios vacíos
- **rm** → Eliminar un archivo
- **touch** → Crear un archivo
- **locate** → Localiza un archivo
- **find** → Ubica archivos dentro de un directorio
- **grep** → Busca a través del texto en un archivo
- **sudo** → Permite realizar tareas de superusuario
- **df** → Informa sobre el uso del disco del sistema
- **du** → Verifica cuánto espacio ocupa un archivo
- **head** → Muestra las primeras líneas de un archivo
- **tail** → Muestra las últimas 10 líneas de un archivo
- **diff** → Compara el contenido de dos archivos
- **tar** → Guarda archivos en formato tarball (.tar)
- **chmod** → Asigna o quita permisos a un archivo
- **chown** → Transfiere la propiedad de un archivo
- **jobs** → Muestra todos los procesos ejecutandose
- **kill** → Finaliza la ejecución de un proceso
- **ping** → Verifica tu conexión a un servidor
- **wget** → Descarga archivos de Internet
- **history** → Ver el historial de comandos
- **man** → Muestra las instrucciones de un comando
- **apt-get** → Instalar / Actualizar un paquete
- **hostname** → Conoce el nombre de tu host/red