Arquitectura Linux

Usuarios, grupos y permisos

¿Qué es un usuario?

- Linux es un sistema multiusuario, es decir, pueden coexistir múltiples usuarios en la misma máquina
- Cada usuario tiene un UID (*User ID*) y uno o varios GID (*Group ID*)
- Existen dos tipos de usuarios: **`usuario estándar`** y **`súper-usuario`**
- Además, se pueden restringir los usuarios según grupo, permisos, etc

Usuario estándar vs súper-usuario

Usuario estándar	Súper-usuario
Tiene acceso a su carpeta personal (`/home/<user>`) y a otras carpetas del sistema, como `/tmp, /run`	Tiene acceso ilimitado a todo el sistema y todos sus recursos
Puede efectuar casi cualquier acción sobre el sistema operativo: instalar programas, añadir aplicaciones, crear servicios, etc	Puede realizar cualquier acción sobre el sistema: instalar una aplicación para todos los usuarios, editar la configuración global del sistema, etc
Tiene la capacidad de crear otros usuarios estándar	Puede crear otros usuarios administradores así como usuarios estándar
Puede cambiar su propia contraseña	Puede cambiar cualquier configuración de cualquier usuario, incluida la contraseña
Puede transferir la propiedad de ficheros/directorios a otro usuario, pero no la puede recuperar	Puede establecer la propiedad de ficheros/directorios hacia cualquier usuario
Tiene acceso, por lo general, completo a casi todos los dispositivos del sistema: discos, USB, etc	Tiene acceso a todos los dispositivos del sistema: desde discos al propio procesador
Puede restringir las propiedades de un fichero/directorio al máximo, de manera que solo él pueda acceder	Puede acceder a cualquier directorio/fichero del sistema, aunque no sea el propietario ni tenga permisos

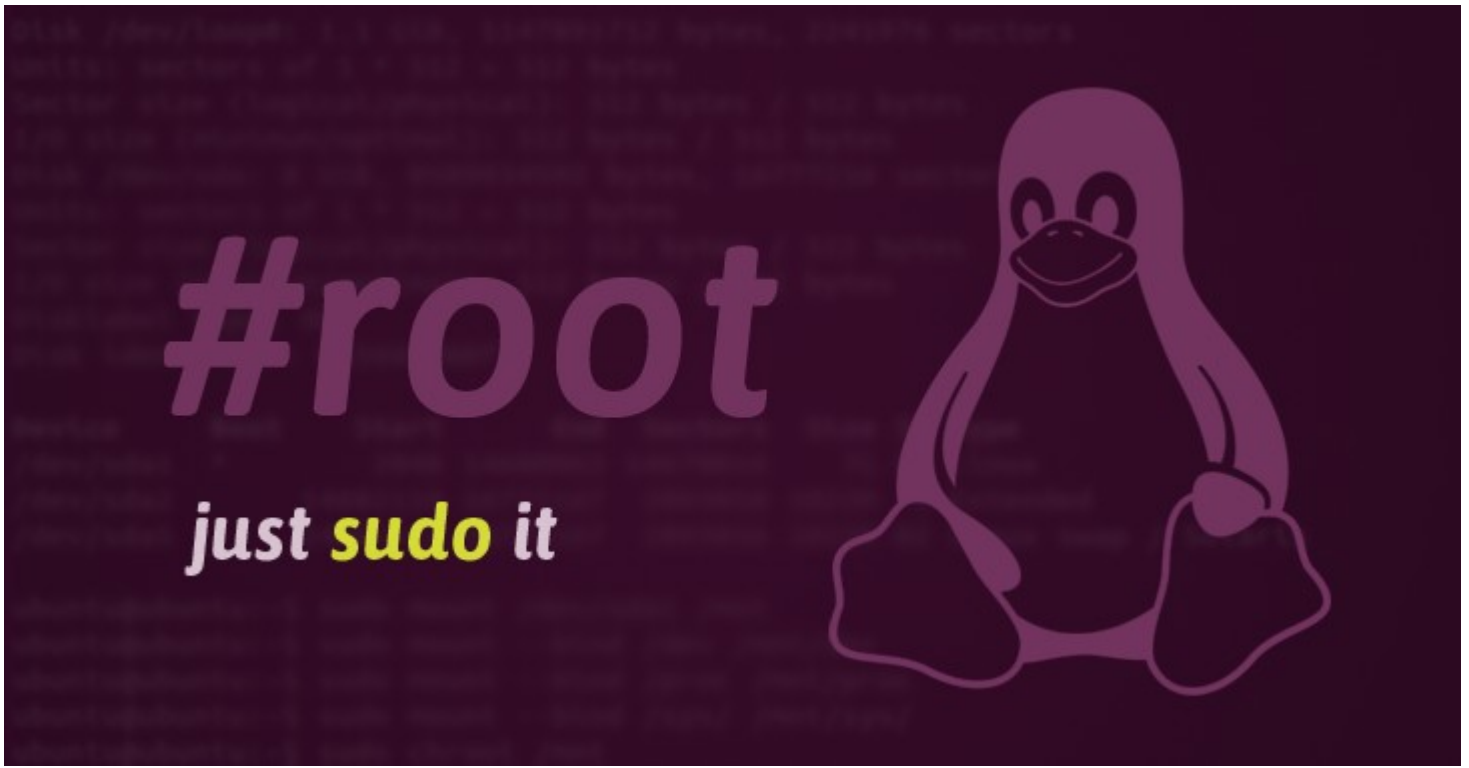
Usuario estándar vs súper-usuario

¿Necesito una cuenta de súper-usuario?

- El súper-usuario suele estar **deshabilitado**
- Se administra el sistema **elevando privilegios**
- Esto permite realizar todas las acciones de forma estándar y administrar el sistema cuando es necesario
- Inclusive, elevando privilegios, hay ciertas acciones que no se podrán hacer y que será necesario acceder como súper-usuario

Usuario estándar vs súper-usuario

¿Necesito una cuenta de súper-usuario?



sudo

super user do...

`sudo <command>`

`sudo -i`

`sudo -u <user> <command>`

`sudo -E <command>`

`sudo -H <command>`

`sudo -e <file>`

```
javinator9889@Tony-MkIII ~ sudo --help
sudo - ejecuta un comando como otro usuario

usage: sudo -h | -K | -k | -V
usage: sudo -v [-AknS] [-g group] [-h host] [-p prompt] [-u user]
usage: sudo -l [-AknS] [-g group] [-h host] [-p prompt] [-U user] [-u user]
            [command]
usage: sudo [-AbEHknPS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p
            prompt] [-T timeout] [-u user] [VAR=value] [-i|-s] [<command>]
usage: sudo -e [-AknS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p
            prompt] [-T timeout] [-u user] file ...
```

Creación de usuarios

useradd (1)

```
javinator9889@Tony-MkIII ▶ ~ ▶ useradd --help

Modo de uso: useradd [opciones] USUARIO
             useradd -D
             useradd -D [opciones]

Opciones:
      --badnames                 do not check for bad names
  -b, --base-dir DIR_BASE        directorio base para el directorio personal
                                 de la nueva cuenta
      --btrfs-subvolume-home     use BTRFS subvolume for home directory
  -c, --comment COMENTARIO       campo GECOS de la nueva cuenta
  -d, --home-dir DIR_PERSONAL    directorio personal de la nueva cuenta
  -D, --defaults                 imprime o cambia la configuración
                                 predeterminada de useradd
  -e, --expiredate FECHA_CADUCIDAD  fecha de caducidad de la nueva cuenta
  -f, --inactive INACTIVO        periodo de inactividad de la contraseña
                                 de la nueva cuenta
  -g, --gid GRUPO                nombre o identificador del grupo primario de
                                 la nueva cuenta
  -G, --groups GRUPOS            lista de grupos suplementarios de la nueva
                                 cuenta
```

Creación de usuarios

useradd (2)

```
  -h, --help                    muestra este mensaje de ayuda y termina
  -k, --skel DIR_SKEL           utiliza este directorio «skeleton» alternativo
  -K, --key CLAVE=VALOR         sobrescribe los valores predeterminados de
                                «/etc/login.defs»
  -l, --no-log-init             no añade el usuario a las bases de datos de
                                lastlog y faillog
  -m, --create-home             crea el directorio personal del usuario
  -M, --no-create-home          no crea el directorio personal del usuario
  -N, --no-user-group           no crea un grupo con el mismo nombre que el
                                usuario
  -o, --non-unique              permite crear usuarios con identificadores
                                (UID) duplicados (no únicos)
  -p, --password CONTRASEÑA     contraseña cifrada de la nueva cuenta
  -r, --system                  crea una cuenta del sistema
  -R, --root CHROOT_DIR         directorio en el que hacer chroot
  -P, --prefix PREFIX_DIR       prefix directory where are located the /etc/* files
  -s, --shell CONSOLA           consola de acceso de la nueva cuenta
  -u, --uid UID                 identificador del usuario de la nueva cuenta
  -U, --user-group              crea un grupo con el mismo nombre que el
                                usuario
  -Z, --selinux-user USUARIO_SE  utiliza el usuario indicado para el usuario
                                 de SELinux
      --extrausers              Use the extra users database
```

Creación de usuarios

useradd (ejemplos)

```
sudo useradd -c "Usuario de prueba" -e 2021/01/20 --no-create-home ISO21

sudo useradd --no-create-home --shell /usr/sbin/nologin Alumno

javinator9889@Tony-MkIII ▶ ~ ▶ useradd -D
GROUP=100
HOME=/home
INACTIVE=-1
EXPIRE=
SHELL=/bin/sh
SKEL=/etc/skel
CREATE_MAIL_SPOOL=no

javinator9889@Tony-MkIII ▶ ~ ▶ cat /etc/passwd| grep -E "(javinator9889|ISO21)"
javinator9889:x:1000:1000:Javinator9889,,,:/home/javinator9889:/usr/bin/zsh
ISO21:x:1001:1002:Usuario de prueba:/home/ISO21:/bin/sh
```

Modificación de usuarios

`usermod (1)`

```
javinator9889@Tony-MkIII ▶ ~ ▶ usermod --help
Modo de uso: usermod [opciones] USUARIO

Opciones:
  -b, --badnames                allow bad names
  -c, --comment COMENTARIO      nuevo valor del campo GECOS
  -d, --home DIR_PERSONAL       nuevo directorio personal del nuevo usuario
  -e, --expiredate FECHA_EXPIR  establece la fecha de caducidad de la
                                cuenta a FECHA_EXPIR
  -f, --inactive INACTIVO       establece el tiempo de inactividad después
                                de que caduque la cuenta a INACTIVO
  -g, --gid GRUPO               fuerza el uso de GRUPO para la nueva cuenta
                                de usuario
  -G, --groups GRUPOS           lista de grupos suplementarios
  -a, --append                  append the user to the supplemental GROUPS
                                mentioned by the -G option without removing
                                the user from other groups
  -h, --help                    muestra este mensaje de ayuda y termina
  -l, --login NOMBRE            nuevo nombre para el usuario
  -L, --lock                    bloquea la cuenta de usuario
  -m, --move-home               mueve los contenidos del directorio
                                personal al directorio nuevo (usar sólo
                                junto con -d)
```

Modificación de usuarios

`usermod (2)`

```
  -o, --non-unique              permite usar UID duplicados (no únicos)
  -p, --password CONTRASEÑA     usar la contraseña cifrada para la nueva cuenta
  -R, --root CHROOT_DIR         directorio en el que hacer chroot
  -P, --prefix PREFIX_DIR       prefix directory where are located the /etc/* files
  -s, --shell CONSOLA           nueva consola de acceso para la cuenta del
                                usuario
  -u, --uid UID                 fuerza el uso del UID para la nueva cuenta
                                de usuario
  -U, --unlock                  desbloquea la cuenta de usuario
  -v, --add-subuids FIRST-LAST  add range of subordinate uids
  -V, --del-subuids FIRST-LAST  remove range of subordinate uids
  -w, --add-subgids FIRST-LAST  add range of subordinate gids
  -W, --del-subgids FIRST-LAST  remove range of subordinate gids
  -Z, --selinux-user SEUSER     new SELinux user mapping for the user account
```

Grupos

Los grupos son una entidad especial dentro de Linux a la cual pueden pertenecer diversos usuarios.

Los grupos permiten definir y restringir el acceso a ciertos recursos del sistema, de manera que es necesario pertenecer a ese grupo para poder usarlos.

```
javinator9889@Tony-MkIII ▶ ~ ▶ groups
javinator9889 adm tty dialout cdrom sudo dip plugdev kvm lpadmin lxd sambashare docker libvirt hugetlbfs
```

https://wiki.debian.org/SystemGroups#Other_System_Groups

Grupos

Un usuario puede ser añadido a un grupo con la orden:

```
sudo usermod -aG group1,group2 username
```

Y crear / eliminar grupos con:

```
sudo groupadd groupname
sudo groupdel groupname
```

Actualmente, un usuario pertenece a un único grupo primario (el que se usa cuando se crea un nuevo fichero) y a $0..n$ grupos secundarios.

```
sudo usermod -g new_primary_group username
```

```
sudo useradd -g primary_group -G group1,group2 username
```

Permisos

Los permisos son una de las funciones más potentes de todo el ecosistema Linux. Permiten, en un servidor / cliente bien configurado, restringir el acceso a los datos e inclusive evitar su cifrado.

A su vez, permisos mal configurados se convierten en el *backdoor* más habitual y peligroso de un sistema Linux.

Los ficheros y directorios en Linux tienen los siguientes permisos:

- Dueño y grupo.
- Modo

Permisos – dueño y grupo

El dueño de un fichero define quién tiene la posesión del mismo. El grupo, por su parte, se usa para conceder ciertos privilegios a los usuarios pertenecientes al mismo.

Sin embargo, hay ciertos usuarios / grupos que tienen privilegios especiales:

- **`root`**, aunque no sea el dueño de un fichero tiene plena potestad sobre el mismo.
- **`sudo`**, donde los usuarios pertenecientes al grupo tienen la posibilidad de realizar acciones como otros usuarios, de forma que se pueden saltar los permisos.

```
javinator9889@Tony-MkIII  /tmp  ls -l example.file
-rw-rw-r-- 1 javinator9889 javinator9889 0 ene 20 15:10 example.file
```

Permisos – dueño y grupo

Se puede cambiar el dueño y el grupo al que pertenece un archivo con el comando **`chown`** (*change owner*).

La sintaxis es:

```
chown user:group [OPTIONS] filename/folder
```

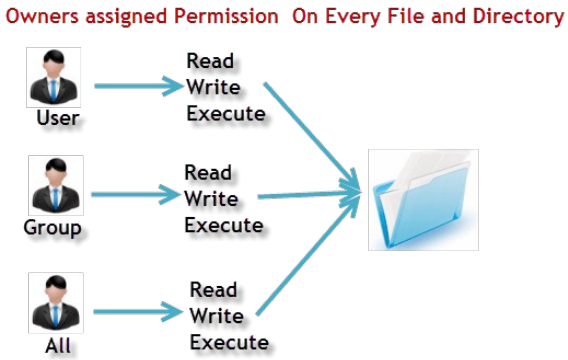
```
chgrp group [OPTIONS] filename/folder
```

```
✘ javinator9889@Tony-MkIII  /tmp  sudo chown www-data:javinator9889 example.file
 javinator9889@Tony-MkIII  /tmp  ls -l example.file
-rw-rw-r-- 1 www-data javinator9889 0 ene 20 15:10 example.file
```

```
 javinator9889@Tony-MkIII  /tmp  sudo chgrp www-data example.file
 javinator9889@Tony-MkIII  /tmp  ls -l example.file
-rw-rw-r-- 1 www-data www-data 0 ene 20 15:10 example.file
```

Permisos – modo

El modo de un fichero / directorio permite configurar quién puede acceder a los datos y bajo qué restricciones. Un modo bien configurado impide a cualquier usuario no autorizado siquiera el acceso a los datos (sería incapaz de saber que están ahí)



El modo se configura para:

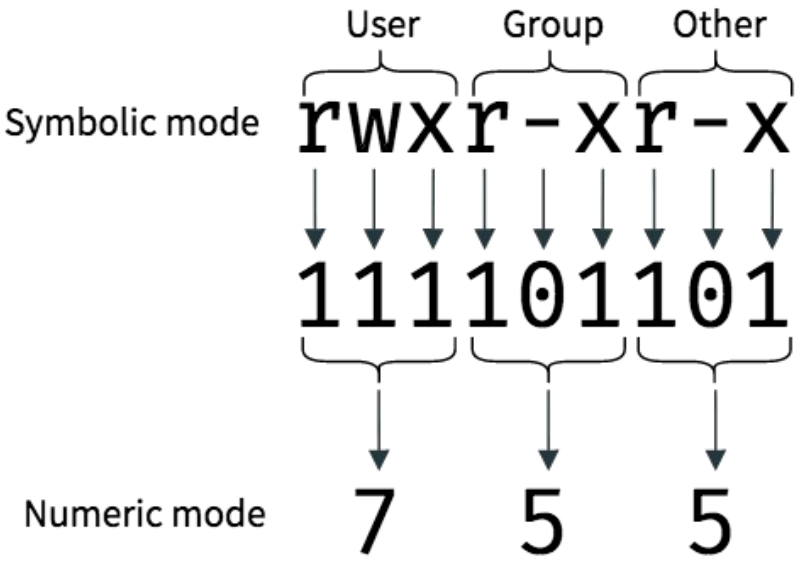
- El dueño
- El grupo
- El resto de usuarios

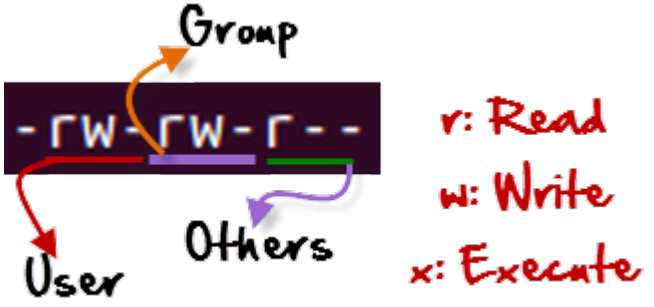
Y se distingue además los siguientes modos:

1. Lectura: en un fichero, la posibilidad de abrirlo y leerlo. En un directorio, la posibilidad de listar su contenido.
2. Escritura: en un fichero, la posibilidad de modificar los contenidos del mismo. En un directorio, la posibilidad de crear, renombrar y eliminar los ficheros.
3. Ejecución: en un fichero, la posibilidad de ejecutar el código fuente que contiene. En un directorio, la posibilidad de poder acceder al mismo y a los ficheros y directorios que contiene.

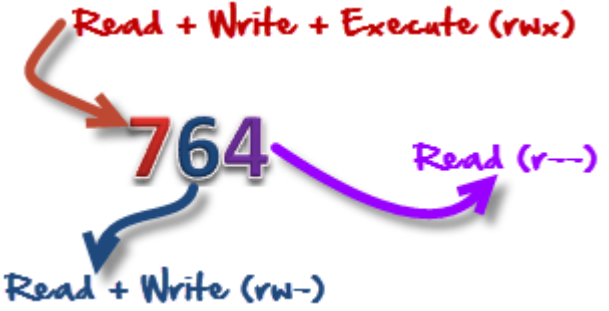
Permisos – modo

El modo se cambia con el comando **`chmod`** (*change mode*), y se puede cambiar de forma semántica o de forma absoluta (numérica)






```
javinator9889@Tony-MkIII  /tmp  chmod 764 example.file
javinator9889@Tony-MkIII  /tmp  ls -l example.file
-rwxrw-r-- 1 javinator9889 javinator9889 0 ene 20 15:10 example.file
```



Permisos – modo

El cambio de forma simbólica se realiza mediante una serie de reglas sintácticas:

+	Añadir el permiso a un fichero / directorio
–	Elimina el permiso
=	Establece el permiso y sobrescribe los anteriores
u	Aplica el cambio de modo al dueño
g	Aplica el cambio de modo al grupo
o	Aplica el cambio de modo a “otros”
a	Aplica el cambio de modo a todos

```
javinator9889@Tony-MkIII  /tmp  chmod o=rwx example.file
javinator9889@Tony-MkIII  /tmp  ls -l example.file
-rwxrw-rwx 1 javinator9889 javinator9889 0 ene 20 15:10 example.file
```

```
javinator9889@Tony-MkIII  /tmp  chmod g+x example.file
javinator9889@Tony-MkIII  /tmp  ls -l example.file
-rwxrwxrwx 1 javinator9889 javinator9889 0 ene 20 15:10 example.file
```

```
javinator9889@Tony-MkIII  /tmp  chmod u-rx example.file
javinator9889@Tony-MkIII  /tmp  ls -l example.file
--w-rwxrwx 1 javinator9889 javinator9889 0 ene 20 15:10 example.file
```

```
javinator9889@Tony-MkIII  /tmp  chmod a+rwx example.file
javinator9889@Tony-MkIII  /tmp  ls -l example.file
-rwxrwxrwx 1 javinator9889 javinator9889 0 ene 20 15:10 example.file
```

```
javinator9889@Tony-MkIII  /tmp  chmod 0644 example.file
javinator9889@Tony-MkIII  /tmp  ls -l example.file
-rw-r--r-- 1 javinator9889 javinator9889 0 ene 20 15:10 example.file
```

Permisos – modo

000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

Establecer el modo de los ficheros usando el código binario permite mayor rapidez y agilidad. Al principio puede resultar más complejo, pero acaba siendo más rápido y eficiente