

En **Linux**, la gestión de permisos para directorios y archivos es fundamental para la seguridad y el control de acceso. Los permisos se configuran mediante los comandos `chmod`, `chown` y `chgrp`.

1. Permisos en Linux

Cada archivo y directorio en Linux tiene tres tipos de permisos para tres categorías de usuarios:

- **Usuarios (User - u)** → El propietario del archivo.
- **Grupo (Group - g)** → Grupo al que pertenece el archivo.
- **Otros (Others - o)** → Todos los demás usuarios.

Permisos básicos

Cada archivo o directorio puede tener los siguientes permisos:

Permiso	Símbolo	Valor numérico	Descripción
Lectura	r	4	Permite leer el contenido de un archivo o listar un directorio.
Escritura	w	2	Permite modificar el archivo o agregar/eliminar archivos dentro de un directorio.
Ejecución	x	1	Permite ejecutar un archivo como un programa o acceder a un directorio.

Ejemplo de representación de permisos

`-rwxr-xr-- 1 usuario grupo 1234 Feb 20 12:34 archivo.txt`

- `rwx` → Usuario (propietario) tiene permisos de lectura, escritura y ejecución.
 - `r-x` → Grupo tiene permisos de lectura y ejecución.
 - `r--` → Otros solo tienen permiso de lectura.
-

2. Cambiar permisos con chmod

Modo simbólico

chmod u+x archivo.txt # Añadir permiso de ejecución al usuario

chmod g-w archivo.txt # Quitar permiso de escritura al grupo

chmod o=r archivo.txt # Establecer solo lectura para otros

chmod u=rwx,g=rx,o= archivo.txt # Configurar permisos exactos

Modo numérico

chmod 755 archivo.txt # rwx para usuario, rx para grupo y otros

chmod 644 archivo.txt # rw para usuario, r para grupo y otros

chmod 700 archivo.txt # rwx solo para usuario

Explicación:

- **7 (rwx) = 4+2+1** (usuario)
 - **5 (r-x) = 4+0+1** (grupo)
 - **5 (r-x) = 4+0+1** (otros)
-

3. Cambiar propietario y grupo

Cambiar propietario

chown usuario archivo.txt

Cambiar grupo

chgrp grupo archivo.txt

Cambiar ambos

chown usuario:grupo archivo.txt

4. Permisos especiales

SUID (Set User ID)

Cuando un archivo tiene el bit **SUID**, se ejecuta con los permisos del propietario en lugar del usuario que lo ejecuta.

```
chmod u+s archivo
```

Ejemplo:

```
-rwsr-xr-x 1 root usuario 1234 Feb 20 12:34 programa
```

El usuario que lo ejecuta tendrá permisos de root (si el propietario es root).

SGID (Set Group ID)

Cuando un directorio tiene **SGID**, los archivos creados dentro heredan el grupo del directorio en lugar del grupo del usuario creador.

```
chmod g+s directorio
```

Ejemplo:

```
drwxr-sr-x 1 usuario grupo 4096 Feb 20 12:34 directorio
```

Sticky Bit

Evita que los usuarios eliminen archivos de un directorio si no son propietarios.

```
chmod +t directorio
```

Ejemplo:

```
drwxrwxrwt 1 usuario grupo 4096 Feb 20 12:34 /tmp
```

Se usa en directorios públicos como /tmp.

5. Ver permisos

Para ver los permisos de un archivo o directorio:

```
ls -l archivo.txt
```

```
ls -ld directorio
```

Resumen rápido

Acción	Comando
Cambiar permisos	<code>chmod 755 archivo.txt</code>
Cambiar propietario	<code>chown usuario archivo.txt</code>
Cambiar grupo	<code>chgrp grupo archivo.txt</code>
Ver permisos	<code>ls -l archivo.txt</code>
Añadir SUID	<code>chmod u+s archivo</code>
Añadir SGID	<code>chmod g+s directorio</code>
Añadir Sticky Bit	<code>chmod +t directorio</code>

Si necesitas más detalles o ejemplos, dime qué caso específico quieres configurar. 🚀