

# Gestión de Usuarios y Permisos

## Windows 11

En Windows 11, la gestión de usuarios y grupos se realiza principalmente a través de la interfaz gráfica y la consola de administración.

### 1. Perfiles de Usuario:

- **Administrador:** Tienen control total sobre el sistema, pudiendo instalar software, modificar configuraciones del sistema, gestionar otros usuarios y acceder a todos los archivos. Son esenciales para el mantenimiento y la configuración inicial del sistema.
- **Usuario Estándar:** Cuentas con permisos limitados, adecuadas para tareas diarias como navegar por internet, usar aplicaciones de oficina y acceder a sus propios documentos. No pueden instalar software que afecte a todo el sistema ni modificar configuraciones críticas.
- **Desarrollador:** Pueden operar como usuarios estándar para tareas básicas, pero necesitan privilegios específicos (posiblemente temporales o con `sudo` en Linux o elevación de permisos en Windows) para instalar herramientas de desarrollo, acceder a directorios de proyecto y ejecutar scripts.

### 2. Creación de Usuarios:

- **Desde PowerShell:**  
PowerShell  
# Crear un nuevo usuario local  
New-LocalUser -Name "NombreUsuario" -Password (ConvertTo-SecureString "ContraseñaSegura123" -AsPlainText -Force) -FullName "Nombre Completo" -Description "Usuario Estándar"
- # Añadir el usuario al grupo de Administradores (si se necesita elevar privilegios)  
Add-LocalGroupMember -Group "Administradores" -Member "NombreUsuario"

### 3. Gestión de Grupos y Permisos:

- **Grupos Locales:** Windows utiliza grupos para asignar permisos de manera eficiente. Los grupos más comunes son "Administradores", "Usuarios", "Usuarios del Escritorio Remoto", etc.
- **Asignación de Usuarios a Grupos:**
  - **Desde PowerShell:**  
PowerShell  
# Añadir un usuario a un grupo existente  
Add-LocalGroupMember -Group "Usuarios" -Member "NombreUsuario"

- **Permisos de Archivos y Carpetas (NTFS):**
  - Hacer clic derecho en la carpeta o archivo > **Propiedades** > **Seguridad**.
  - Desde aquí, se pueden ver los permisos actuales y modificar los permisos para usuarios o grupos específicos (Control total, Modificar, Leer y ejecutar, Leer, Escribir).

## Rocky Linux

En Rocky Linux, la gestión de usuarios y permisos se realiza principalmente a través de la línea de comandos, utilizando comandos como **useradd**, **usermod**, **groupadd**, **chmod** y **chown**.

### 1. Perfiles de Usuario:

- **Administrador (o root / usuarios sudo):** El usuario **root** tiene control absoluto. Para tareas administrativas, se recomienda usar un usuario estándar que pertenezca al grupo **wheel** (o **sudoers**), y usar **sudo** para ejecutar comandos con privilegios elevados.
- **Usuario Estándar:** Cuentas con privilegios básicos para acceder a sus propios directorios y ejecutar aplicaciones.
- **Desarrollador:** Necesitarán acceso a herramientas específicas y directorios de proyectos, lo que puede implicar pertenecer a grupos adicionales o tener permisos de escritura en ciertos directorios.

### 2. Creación de Usuarios:

#### Crear un nuevo usuario estándar:

Bash

```
sudo useradd -m nombre_usuario # -m crea el directorio home del usuario
sudo passwd nombre_usuario # Establece la contraseña para el nuevo usuario
```

○

#### Crear un usuario con privilegios de administrador (sudo):

Bash

```
sudo useradd -m admin_usuario
sudo passwd admin_usuario
sudo usermod -aG wheel admin_usuario # Añadir al grupo wheel para permisos sudo
```

○

### 3. Gestión de Grupos:

#### Crear un nuevo grupo:

Bash

```
sudo groupadd nombre_grupo
```

○

#### **Añadir un usuario a un grupo existente:**

Bash

sudo usermod -aG nombre\_grupo nombre\_usuario # -aG añade al grupo sin eliminar los grupos existentes

○

#### **Eliminar un usuario de un grupo:**

Bash

sudo gpasswd -d nombre\_usuario nombre\_grupo

○

#### **Ver los grupos a los que pertenece un usuario:**

Bash

groups nombre\_usuario

○

### **4. Permisos de Archivos y Directorios (Sistema de Permisos de Linux):**

- Linux utiliza un sistema de permisos basado en tres tipos de usuarios (propietario, grupo y otros) y tres tipos de permisos (lectura **r**, escritura **w**, ejecución **x**).

#### **Visualizar permisos:**

Bash

ls -l /ruta/al/archivo\_o\_directorio

○

- **Cambiar permisos (chmod):**

- **Formato numérico (octal):**

- **r=4, w=2, x=1**

- Ejemplo: **chmod 755 archivo.sh** (propietario: rwx, grupo: rx, otros: rx)

- **chmod 644 archivo.txt** (propietario: rw, grupo: r, otros: r)

- **Formato simbólico:**

- **u** (propietario), **g** (grupo), **o** (otros), **a** (todos)

- Ejemplo: **chmod u+x archivo.sh** (añadir permiso de ejecución al propietario)

- **chmod go-w archivo.txt** (eliminar permiso de escritura para grupo y otros)

- **Cambiar propietario y grupo (chown, chgrp):**

#### **Cambiar propietario:**

Bash

sudo chown nuevo\_propietario /ruta/al/archivo\_o\_directorio

- 

**Cambiar propietario y grupo:**

Bash

```
sudo chown nuevo_propietario:nuevo_grupo /ruta/al/archivo_o_directorio
```

- 

- **Cambiar solo el grupo:**

Bash

```
sudo chgrp nuevo_grupo /ruta/al/archivo_o_directorio
```