

Capítulo 7

Permisos de acceso a los archivos

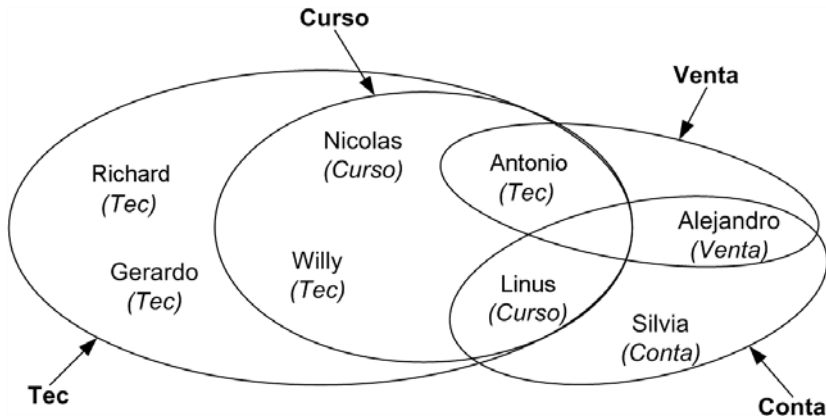
1. Conceptos de cuentas de usuario y de grupos

El sistema GNU/Linux es multiusuario, por tanto las personas que lo usan deben identificarse para asegurar la confidencialidad de los datos contenidos en los archivos. En efecto, no sería aceptable que el usuario "Nicolas" pudiera consultar los archivos personales de "Richard" sin el permiso de este.

Cada una de estas personas dispone por tanto de una "cuenta de usuario" en el sistema; pueden utilizarlo tras ser claramente identificadas. Sin embargo, está permitido compartir archivos entre colaboradores y existe una noción de "grupo de usuarios" en GNU/Linux.

Un usuario debe ser miembro obligatoriamente de un grupo de usuarios en un sistema Unix como GNU/Linux: su grupo principal es el utilizado al crear archivos. Por el contrario, puede pertenecer a otros varios grupos: sus grupos secundarios determinan sus derechos de acceso a los archivos creados por otros miembros de los grupos.

Por ejemplo, si se representa los diferentes servicios de una empresa con su personal, puede observarse que, si bien cada individuo tiene una función primaria (indicada entre paréntesis), algunos pueden asumir varias misiones:



Se observa aquí que:

- Richard y Gerardo pertenecen al servicio técnico (Tec).
- Nicolas, que es ante todo formador (Curso), también forma parte del servicio técnico (Tec).
- Willy, que pertenece al servicio técnico (Tec) principalmente, también trabaja en el departamento de formación (Curso).
- Linus es un formador (Curso) que colabora con los departamentos técnicos (Tec) y de contabilidad (Conta).
- Antonio, del servicio técnico (Tec), ofrece sus competencias al servicio comercial (Venta) y también da clases (Curso).
- Alejandro es un comercial (Venta) que se ocupa también de tareas administrativas (Conta).
- Silvia forma parte únicamente del departamento de contabilidad (Conta).

Para identificar a todos estos usuarios a nivel del sistema operativo, se les atribuye un número único: el UID (*User's ID*); el propietario de un archivo se determina por este número en Unix. Estos usuarios están dotados también de un nombre de usuario único (*login*) y de una contraseña (*password*) para que puedan autenticarse al conectarse al sistema.

De la misma manera, los grupos de usuarios se representan por un nombre único al que se asocia un identificador único: el GID (*Group ID*). Este número se utiliza también para determinar el grupo propietario de un archivo.

1.1 Jerarquía de usuarios

Los usuarios, y por consiguiente las cuentas de usuario, no son todas iguales en Unix. Se pueden distinguir tres tipos de cuentas:

root

Es el usuario más importante del sistema desde el punto de vista de la administración. No se ve afectado por los derechos de acceso a los archivos y puede hacer más o menos de todo en el sistema, excepto escribir en un sistema de archivos montado en lectura exclusiva (CD-ROM). Su UID igual a 0 le confiere su especificidad. Este "superusuario" se encarga de las tareas administrativas del sistema. Para evitar errores al trabajar, es muy recomendable utilizar la cuenta administrativa sólo para las tareas que requieren los derechos de superusuario.

bin, daemon, sync, apache...

Existe en el sistema una serie de cuentas que no se asignan a personas físicas. Estas cuentas sirven para facilitar la administración de los derechos de acceso de ciertas aplicaciones y demonios. Los UID comprendidos entre 1 y 999 se utilizan generalmente para estas cuentas.

linus, nicolas...

Todas las demás cuentas de usuario se asocian a personas reales; su función es permitir a los usuarios estándar conectarse y utilizar los recursos del equipo. El UID de un usuario es normalmente un número superior o igual a 1000.

Observación

Se denominan "demonios" los programas que se ejecutan como tarea en segundo plano, como un servidor web o un servidor de impresión.

Al igual que las cuentas de usuario, existen diferentes tipos de grupos en un sistema GNU/Linux que permiten dar derechos comunes a una serie de usuarios:

root

Su GID es 0 y es el grupo principal del administrador.

bin, daemon, sync, apache...

Estos grupos tienen la misma función que las cuentas del mismo nombre y permiten dar los mismos derechos de acceso a una serie de aplicaciones. Por convención, los grupos del sistema tienen un GID comprendido entre 1 y 999.

curso, tec...

Estos grupos representan una serie de personas reales que deben acceder a los mismos archivos. Típicamente, tienen un GID superior o igual a 1000.

1.2 Comandos útiles

Los comandos **id** y **groups** permiten mostrar información sobre los grupos. El primero da el UID del usuario, el GID de su grupo principal y los GID de todos los grupos a los que pertenece. El segundo sólo proporciona la lista completa de los grupos pero acepta varios nombres de usuario como argumentos:

```
[nicolas]$ whoami
nicolas
[nicolas]$ id
uid=1000(nicolas) gid=1000(curso) grupos=1000(curso),1001(tec)
[nicolas]$ id richard
uid=1002(richard) gid=1001(tec) grupos=1001(tec)
[nicolas]$ groups
curso tec
[nicolas]$ groups gerardo alejandro willy root
antonio : tec curso venta
alejandro : venta conta
willy : tec curso
root : root
```

2. Permisos de Unix

Los permisos de acceso a los archivos determinan las acciones que pueden emprender los usuarios.

■ Observación

La mayoría de los problemas de instalación, configuración y funcionamiento de las aplicaciones en GNU/Linux se debe a derechos de acceso mal adjudicados.

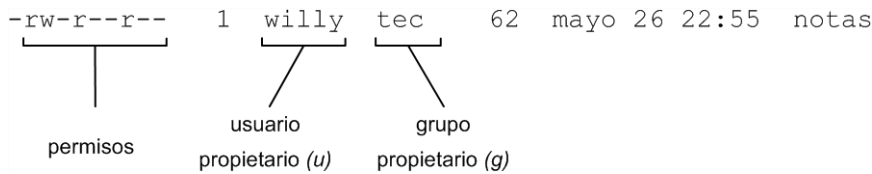
En primer lugar, es necesario saber que los derechos de acceso en Linux se definen por:

- Una cuenta de usuario: propietario del archivo, es en principio el usuario que lo ha creado.
- Un grupo: este grupo es generalmente el grupo principal del propietario del archivo, pero puede ser modificado por este y tomar el valor de uno de sus grupos secundarios.
- Los otros: esta entidad representa toda persona distinta del propietario y que no es miembro del grupo citado anteriormente.

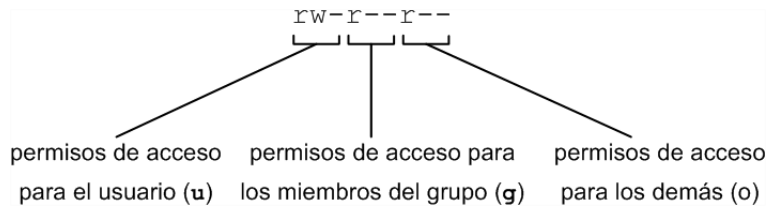
■ Observación

Los derechos de acceso a un archivo se llaman también "modos" en Unix.

Los derechos, el usuario y el grupo propietarios de un archivo pueden verse con el comando **ls -l**:



En este último ejemplo, el archivo pertenece al usuario **willy** y al grupo **tec**; los nueve caracteres **rw-r--r--** definen los derechos de acceso a este archivo para el usuario **willy** (user o **u**), los miembros del grupo **tec** (group o **g**) y los demás (other u **o**). Más exactamente, estos caracteres se distribuyen así:



Todo usuario está asociado, pues, a una de estas entidades para determinar los permisos vigentes.

■ Observación

Atención: si el usuario es propietario del archivo, se aplican los permisos del propietario, y no los del grupo, aunque el usuario sea también miembro de ese grupo.

El comando GNU **ls** puede añadir un carácter adicional a la sucesión de nueve derechos Unix estándar cuando las autorizaciones especiales están ubicadas. Un punto **'** señala un contexto de seguridad SELinux específico y un **+** indica que se utiliza otro método de autorización, como las ACL (*Access Control Lists*).

2.1 Permisos estándar

Los permisos de acceso fundamentales en los archivos y directorios en Unix/Linux son los permisos de lectura **r** (*Read*), escritura **w** (*Write*) y ejecución **x** (*eXecute*).

Estos permisos –definidos para las entidades **u**, **g** y **o**– aparecen en el orden **r**, seguido de **w**, seguido de **x** con el comando **ls -l**. Cuando uno de estos caracteres se reemplaza por un guión, significa que el permiso asociado no está otorgado.

En el ejemplo del apartado anterior, el usuario **willy** que tiene los permisos **rw-**:

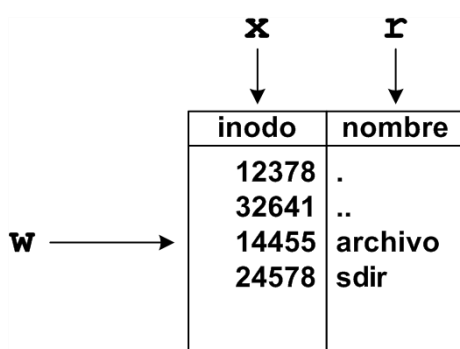
- tiene permiso para leer el archivo *notas*.
- tiene permiso para modificar el archivo *notas*.
- no tiene permiso para ejecutar el archivo *notas*.

Más exactamente, se distinguen los permisos Unix estándar según el tipo de archivo: archivo normal o directorio.

Permiso	Archivo	Directorio
r	Permiso de lectura del contenido del archivo.	Permiso para listar las entradas del directorio.
w	Permiso para modificar el contenido del archivo.	Permiso para modificar las entradas del directorio.
x	Permiso para ejecutar el archivo.	Permiso de acceso a las entradas del directorio.

Es relativamente simple comprender los permisos correspondientes cuando se otorgan a un archivo normal; esto no es tan evidente cuando se trata de un directorio.

Si consideramos los directorios como tablas que contienen, en una columna, los inodos y en otra, los nombres de archivo presentes en el directorio, es más fácil comprender los permisos estándar:



Capítulo 8

Descubriendo el entorno de trabajo

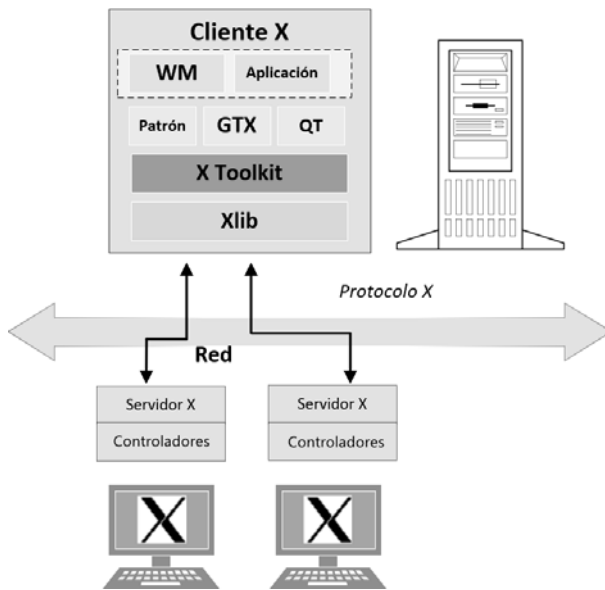
1. Xorg

1.1 Presentación

1.1.1 X Window

Linux dispone de una arquitectura gráfica de base llamada X Window System, comúnmente X Window, X11 o simplemente X. Es un sistema gráfico completo a cargo de diseñar los componentes habituales de un ambiente gráfico de usuario o GUI (*Graphical User Interface*): ventanas, botones, menús, listas, barras de desplazamiento, casillas de verificación, cursor del ratón, etc., y de gestionar los eventos.

X se construye siguiendo una arquitectura del tipo cliente/servidor. Un cliente X es un programa capaz de dialogar con el servidor X. En realidad, un cliente X es un programa gráfico. Para poder comunicarse con el servidor, el cliente utiliza un componente llamado Xlib. El cliente y el servidor no están siempre en la misma máquina. El servidor que gestiona la visualización puede estar en un primer ordenador, mientras que el programa gráfico estará en otro. Las órdenes, llamadas peticiones, entre el cliente y el servidor pasan, entonces, por la red.



No confunda X Window con Windows. El primero es un sistema de visualización cliente/servidor, mientras que el segundo es un sistema operativo. Windows y X Window no son compatibles, aunque es posible instalar un servidor X en Windows.

Si se ejecuta X Window solo, se arranca únicamente el servidor. El resultado puede sorprenderle: una pantalla gris con una cruz gris como cursor del ratón. Por mucho que lo intente, ninguna ventana dará vida al despliegue y los botones del ratón no tendrán ningún efecto. Aunque ejecute un cliente, rápidamente se dará cuenta de un problema: no hay contornos en las ventanas.

1.1.2 El gestor de ventanas

Otro programa cliente X debe informar al servidor cómo diseñar la ventana: es el administrador de ventanas o *Window Manager*. El servidor X muestra el resultado de lo que diseña este gestor: ventanas, selecciones, movimientos y decoraciones (estilos, colores, etc.).

No hay solo un gestor de ventanas, sino varios. Algunos son muy simples y básicos y se limitan al mínimo estricto. Otros, como WindowMaker, son muy completos y permiten trabajar en condiciones realmente cómodas porque, además de las ventanas básicas, ofrecen temas visuales agradables y personalizables, menús contextuales y a veces incluso paneles de configuración. Durante mucho tiempo, estos gestores de ventanas han sido los más utilizados porque son rápidos, eficientes y consumen pocos recursos.

Unity utiliza el administrador Compiz, Gnome emplea Mutter y Metacity, KDE maneja Kwin, Xfce usa Xfwm4...

1.1.3 Los widgets y los toolkits

Cada WM (*Window Manager* o gestor de ventanas) o programa usa bibliotecas gráficas que proponen funciones diseñadas para crear elementos de interfaces gráficas. Un elemento de la interfaz gráfica (botón, menú, campo de entrada, etc.) se denomina **widget** (contracción de *window gadget*). Cuando la biblioteca contiene un kit completo y extenso de widgets, se le llama *Widget Toolkit*, o simplemente **toolkit** o **caja de herramientas para interfaz gráfica**.

Existen diversos toolkits. X Window dispone de un toolkit predefinido llamado Xt. El más conocido desde hace tiempo ha sido MOTIF. Como durante mucho tiempo se consideró propietario (no libre hasta 2000), los programadores crearon otros, como GTK o Qt.

GTK (GIMP toolkit) fue creado para diseñar la interfaz gráfica del programa de edición de imágenes GIMP. A medida que se hizo más y más potente, muchos programadores lo adoptaron para sus propios programas y GTK se independizó. Hoy en día es la biblioteca predefinida del entorno ofimático de GNOME.

Qt (pronunciado «cute», que significa «lindo» en español) es otro toolkit popular. Ofrece, además de las funciones de dibujo y gestión de widgets, un entorno completo para el desarrollo de aplicaciones gráficas y no gráficas: bases de datos SQL, XML, multithreading, gestión de archivos, internacionalización, etc. La biblioteca Qt es la que utiliza el entorno ofimático KDE.

1.1.4 Wayland

Wayland es otro protocolo de servidor de despliegue utilizado en la versión 17.10 de Ubuntu como reemplazo de Xorg (este último se considera «envejecido», ya que data de los años 80...). Estaba presente de forma predefinida, aunque Xorg no había sido suprimido.

Contrariamente a Xorg, Wayland toma en cuenta la seguridad de la información durante la navegación entre las ventanas y las aplicaciones. Limita el código ejecutado como administrador, mejora el rendimiento y la representación gráfica (compositor 3D como servidor de visualización principal).

Sin embargo, un gran número de aplicaciones no funcionan bajo Wayland. Además de ello, el uso compartido de la pantalla y el control de escritorio remoto (RDP, *Remote Desktop Control*) funcionan mejor con Xorg. Por lo tanto, Canonical ha decidido volver a Xorg en esta última versión de Ubuntu. Wayland queda disponible como opción. En la pantalla de inicio de sesión, basta con hacer clic en la rueda dentada a lado del botón **Conectarse** y seleccionar **Ubuntu en Wayland**.

Wayland continúa desarrollándose y mejorándose; sigue siendo una opción viable para las próximas versiones de Ubuntu.

1.2 Instalación y pruebas

Si instaló una versión de Ubuntu Desktop, puede saltarse esta etapa.

1.2.1 Instalar Xorg

El entorno gráfico no se instala de forma predefinida en Ubuntu Server. Además, en un entorno virtual (VirtualBox, VMware...), el servidor Xorg no es necesario.

■ Se debe pasar por apt:

```
■ # apt install xorg
```

Esto descarga e instala alrededor de 200 paquetes de Ubuntu 20.04 (alrededor de 400 MB).

▣ Reinicialice la máquina. El modo de inicio cambia automáticamente a modo gráfico (nivel de ejecución u objetivo `systemd`). Llegará a una ventana de inicio de sesión clásica, como en la versión Desktop. Un entorno de escritorio, incluyendo un gestor de ventanas, se instala y configura automáticamente. A partir de ahora tendrá acceso a su servidor a partir de un entorno gráfico. Pero el entorno está desnudo, con muy pocas aplicaciones instaladas.

▣ Si alguna vez tiene necesidad de arrancar la interfaz gráfica manualmente, use el siguiente comando:

```
■ # systemctl isolate graphical.target
```

o bien:

```
■ # init 5
```

▣ El entorno gráfico aparece en la terminal 1 (`tty1`). Puede volver al modo consola a partir de la terminal 3, por ejemplo.

▣ Para cambiar el estado del sistema y pasar al modo consola, utilice el siguiente comando:

```
■ # systemctl isolate multi-user.target
```

o bien:

```
■ # init 3
```

Atención: la sesión gráfica se cerrará. Recuerde guardar los cambios si es necesario. La terminal 1 pasará ahora a modo consola.

Los comandos relativos a `systemctl` se explicaron en el capítulo Configuración del sistema.

1.2.2 Instalar un gestor de ventanas

Con el servidor X se instaló automáticamente un gestor de ventanas. Puede perfectamente instalar otro; Fluxbox, por ejemplo, de tamaño reducido.

```
■ # apt install fluxbox
```

Si se utiliza Ubuntu como una simple terminal X, este tipo de entorno es suficiente. El botón derecho del ratón permite acceder a un menú para lanzar las distintas aplicaciones, pero en este tipo de entorno gráfico se utilizan principalmente terminales como **xterm**.

Puede probar este entorno seleccionando **fluxbox** en la ventana de acceso.

1.2.3 Instalar un entorno de escritorio

Para trabajar en un entorno agradable, probablemente prefiera utilizar uno de escritorio, un **desktop**. El entorno oficial de Ubuntu es GNOME acompañado de Unity, pero no es el único:

```
$ apt-cache search ubuntu-desktop
ubuntu-desktop - The Ubuntu desktop system
ubuntu-desktop-minimal - The Ubuntu desktop minimal system
kubuntu-desktop - Kubuntu Plasma Desktop/Netbook system
lubuntu-desktop - Lubuntu Desktop environment
xubuntu-desktop - Xubuntu desktop system
```

- **ubuntu-desktop**: es el entorno predefinido que se instala con la versión Desktop de Ubuntu. Incluye GNOME.
- **ubuntu-desktop-minimal**: es el entorno mínimo, sin aplicaciones de terceros. Solo se instalan las aplicaciones propias de Ubuntu.
- **kubuntu-desktop**: escritorio KDE4, basado en Qt.
- **lubuntu-desktop**: escritorio basado en LXQt, de tamaño aún más reducido; puede preferirse en máquinas más antiguas.
- **xubuntu-desktop**: escritorio basado en XFCE, un entorno completo y de tamaño reducido.

Puede instalar varios entornos, de modo que podrá elegir cuál quiere utilizar en el momento de realizar la conexión.

Observación

¡Atención! Los paquetes del entorno gráfico pueden ocupar mucho espacio: más de 1 GB en disco para ubuntu-desktop.

▣ Instale **ubuntu-desktop** y espere mientras se descargan e instalan centenares de paquetes:

```
■ # apt install ubuntu-desktop
```

▣ Finalmente, cierre el entorno gráfico actual activando una consola ([Ctrl][F3]) y reiniciando el servicio que gestiona el escritorio, probablemente gdm3:

```
■ # systemctl restart gdm3
```

1.3 Configuración detallada del servidor Xorg

1.3.1 Generar automáticamente un archivo xorg.conf

El servidor X se configura automáticamente cuando la máquina arranca. La utilización de un archivo de configuración se justifica solo en el caso de una configuración especial (dos tarjetas gráficas, tarjeta erróneamente detectada, pantalla mal conectada...), que exige una mayor precisión en función de los controladores de los periféricos.

El comando **Xorg** puede generar automáticamente un archivo de configuración en función de su equipo. Así, este archivo puede servir de base para aportar todas sus modificaciones.

```
■ # Xorg -configure
```

El archivo generado se coloca, de forma predefinida, en **/root/xorg.conf.new**.

▣ Puede probarlo ejecutando **Xorg** con esta configuración:

```
■ Xorg -config /root/xorg.conf.new
```

▣ Si el archivo le conviene, cópielo en **/etc/X11** para usarlo de forma definitiva.

```
■ # cp /root/xorg.conf.new /etc/X11/xorg.conf
```

En caso de problema, borre dicho archivo. En cada ejecución, **Xorg** detecta por sí mismo los parámetros y establece una configuración correcta.