



Tecnicatura Universitaria en Software Libre



Administración de GNU/Linux I

Unidad 3
Usuarios y Grupos

Autor
Leonardo Martinez

Contenido

1 Usuarios y grupos	3
1.1 Bases de datos de usuarios	4
1.1.1 Creación de usuarios	6
1.2 Bases de datos de grupos	8
1.2.1 Creación de grupos	10
1.2.2 Agregar usuarios a grupos	10
1.2.3 Eliminación de usuarios y grupos	10
2 Más herramientas de gestión de usuarios y grupos	13
2.1 El comando <code>id</code>	13
2.2 El comando <code>newusers</code>	14
2.3 El comando <code>passwd</code>	15
3 Bibliografía	16

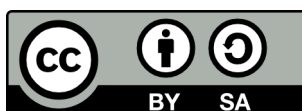
Edición 2022.

Autor: Leonardo Martinez

¡Copia este texto!

Los textos que componen este trabajo se publican bajo formas de licenciamiento que permiten la copia, la redistribución y la realización de obras derivadas, siempre y cuando éstas se distribuyan bajo las mismas licencias libres y se cite la fuente. El copyright de los textos individuales corresponde a los respectivos autores.

Este trabajo está licenciado bajo un esquema Creative Commons Atribución Compartir Igual (CC-BY-SA) 4.0 Internacional. <http://creativecommons.org/licenses/by-sa/4.0/deed.es>



1 Usuarios y grupos

Una característica importante de los sistemas Gnu/Linux es que son sistemas operativos multiusuario, esto permite que más de un usuario pueda realizar tareas con el sistema al mismo tiempo, logrando además que los trabajos concurrentes no interfieran unos con otros en el uso de los recursos.

Los sistemas *GNU/Linux*, al igual que todos los sistemas del tipo *Unix*, tienen identificados dos tipos o categorías de usuarios, diferenciadas por el tipo de función y roles dentro del sistema.

El usuario que tiene todos los permisos sobre el sistema es denominado el **superusuario**, generalmente identificado como el usuario local `root`. Los usuarios restantes con el rol de **Usuario Estándar**, tienen un espacio dedicado en el directorio `/home/<nombre_usuario>` dentro del cual pueden gestionar su espacio de trabajo con total libertad.

Estos espacios individuales son exclusivos de cada usuario y cuyos permisos no permiten el acceso por parte de otros usuarios del sistema, con la única excepción del usuario `root`.

Fuera del directorio personal dentro de `/home` los usuarios pueden realizar tareas simples en el resto del sistema de archivo de *GNU/Linux*, en algunos casos sólo de lectura y ejecución. La gestión de acceso a los archivos para realizar estas tareas se formaliza mediante el otorgamiento de permisos sobre archivos y directorios.

Se pueden diferenciar tres tipos de cuentas de usuario en un sistema GNU/Linux:

- **Cuenta del Administrador:** es la cuenta del usuario que se utiliza para las operaciones de administración del sistema. Identificado con el usuario local `root` es el que más permisos posee, tiene acceso completo a la computadora y los archivos de configuración. Esta capacidad marca también la responsabilidad con la que debe utilizarse el usuario ya que los errores que se cometen con este usuario pueden resultar en problemas graves. Se recomienda que no se utilice como un usuario normal, sino en casos de tareas de administración del sistema.
- **Cuentas de usuarios:** son las cuentas de los usuarios que realizan sus tareas diarias con el sistema, tienen permisos restringidos al directorio personal, algunas áreas del sistema como los archivos temporales en `/tmp` y a los dispositivos habilitados.
- **Cuentas especiales de los servicios:** Son las cuentas que están asociadas a servicios del sistema como `lp`, `www-data` o `mysql`. Hay algunos servicios que corren con el usuario `root` y otros que se pueden configurar para que lo hagan, aunque lo más recomendable es utilizar el usuario específico del servicio.

Es recomendable elevar los privilegios de usuarios normales, aunque generalmente se utiliza uno, para realizar tarea de administración del sistema sin usar el usuario `root`.

Esta configuración permite que usuarios normales ejecuten comandos con privilegios de administrador utilizando el comando `sudo`.

Durante el proceso de instalación de algunas distribuciones se permite bloquear el acceso con el usuario `root` y configurar un usuario normal que será identificado como avanzado otorgándole los privilegios de administrador.

Una vez finalizada la instalación cualquier usuario puede elevarse al rango de administrador agregándolo al grupo `sudo`.

1.1 Bases de datos de usuarios

La lista de *usuarios* del sistema se encuentra en el archivo de configuración `/etc/passwd` y las contraseñas cifradas se guardan en el archivo `/etc/shadow`. Ambos son archivos de texto plano.

Por ejemplo, el archivo `/etc/passwd` de la máquina virtual que se utiliza de ejemplo es:

```
$ cat /etc/passwd

root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/bin/false
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd/netif:/bin/false
systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd/resolve:/bin/false
systemd-bus-proxy:x:103:105:systemd Bus Proxy,,,:/run/systemd:/bin/false
syslog:x:104:108:./home/syslog:/bin/false
_apt:x:105:65534:./nonexistent:/bin/false
messagebus:x:106:110:./var/run/dbus:/bin/false
uidd:x:107:111:./run/uidd:/bin/false
lightdm:x:108:114:Light Display Manager:/var/lib/lightdm:/bin/false
whoopsie:x:109:116:./nonexistent:/bin/false
avahi-autoipd:x:110:119:Avahi autoip daemon,,,:/var/lib/avahi-autoipd:/bin/false
avahi:x:111:120:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/bin/false
colord:x:112:123:colord colour management daemon,,,:/var/lib/colord:/bin/false
dnsmasq:x:113:65534:dnsmasq,,,:/var/lib/misc:/bin/false
hplip:x:114:7:HPLIP system user,,,:/var/run/hplip:/bin/false
kernoops:x:115:65534:Kernel Oops Tracking Daemon,,,:/bin/false
pulse:x:116:124:PulseAudio daemon,,,:/var/run/pulse:/bin/false
rtkit:x:117:126:RealtimeKit,,,:/proc:/bin/false
saned:x:118:127:./var/lib/saned:/bin/false
usbmux:x:119:46:usbmux daemon,,,:/var/lib/usbmux:/bin/false
```

```
speech-dispatcher:x:120:29:Speech Dispatcher,,,:/var/run/speech-dispatcher:/bin/false
administrador:x:1000:1000:Administrador,,,:/home/administrador:/bin/bash
sshd:x:121:65534:/:/var/run/sshd:/usr/sbin/nologin
```

Cada registro del archivo corresponde a un usuario del sistema, sea este administrador, usuario normal o de servicio.

El registro es una serie de campos separados por el caracter `:` con la siguiente información:

Nombre de usuario

Es la cadena de caracteres que se utiliza como identificación del usuario para el *login* o ingreso al sistema, y luego para asociarlo a todos los procesos que ejecute el usuario durante sus sesión de trabajo. Se puede obtener durante la sesión con el comando `whoami` o consultando las variables de entorno `$USER` y `$LOGNAME`.

Contraseña

Es una cadena de caracteres, resultado del cifrado de la cadena mediante una función unidireccional (generalmente *crypt* o *md5*) que ingresa el usuario al momento de definirla. Si se utiliza el método **shadow**, en este campo aparece el caracter `x` y la cadena cifrada se almacena en el archivo de texto `/etc/shadow`.

UID (User IDentifier number)

El número de identificador de usuario, único para cada usuario del sistema. Se puede obtener consultando la variable de entorno `$UID`.

GID (Group IDentifier number)

El número de identificador de grupo primario del usuario. En las distribuciones *GNU/Linux* que crean un grupo propio para cada usuario, como *Debian*, el **GID** coincide con el **UID**.

GECOS (General Electric Comprehensive Operating System)

Es un campo de datos múltiples acerca del usuario. Son cinco cadenas de caracteres separadas por el caracter `,`. El primero es el nombre completo del usuario, y los otros, que no son generalmente utilizados, corresponden al número o descripción de la oficina, al teléfono de la oficina, el teléfono de la casa y el último corresponde a otros o información accesorio. Es un campo heredado del uso en los centros de cómputos con terminales, distribuidas por edificios y campus, conectadas al servidor.

Directorio de inicio de sesión

Es el directorio asignado al usuario para almacenar y trabajar con sus archivos personales. La variable de entorno `$HOME` contiene el paso absoluto a este directorio, generalmente con la estructura `/home/<Nombre_de_usuario>`.

Programa a ejecutar en el inicio de sesión

Es el primer programa que se ejecuta en la sesión del usuario, generalmente es un intérprete de comandos del tipo consola. Por defecto en la mayoría de las distribuciones *GNU/Linux* se utiliza el intérprete `/bin/bash`. Si se especifica `/bin/false`, el usuario no podrá iniciar sesión.

Como se ve en el listado, hay usuarios de servicios y usuarios locales, el usuario `root` tiene el **UID** 0 y el usuario `administrador` el **UID** 1000. Este es el primer número disponible para los usuarios normales del sistema, a partir del cual se van asignando a los nuevos usuarios en forma incremental.

Los registros para los usuarios `root` y `administrador` del archivo de texto `/etc/shadow` son:

```
root:!:16933:0:99999:7:::
administrador:$6$wukE191x$tKy3Jvaloz.d1YoS4KPsAlgFO4KlelP8B1evBFv77e6wUGn8j9X0Zc5
DMIuVjO5GGMgWYeb4eK4AySP9Us55G1:16933:0:99999:7:::
```

En el primer campo se encuentra el identificador del usuario y en el segundo la contraseña cifrada. El carácter `!` en el campo de la contraseña indica que el inicio de sesión con este usuario está deshabilitado. El resto de los campos indican:

- Días desde el 1 de enero de 1970 en que la contraseña se cambió por última vez.
- Días que faltan para que se cambie, si se indica `0` no hay que cambiarla.
- Días después en que hay que cambiarla, es decir el plazo de cambio.
- Días en que el usuario será avisado antes de que la contraseña expire.
- Días, una vez expirado, en que se producirá la deshabilitación de la cuenta.
- Días desde el 1 de enero de 1970 en que la cuenta está deshabilitada.
- Y el último es un campo reservado.

1.1.1 Creación de usuarios

Una de las tareas más importantes del administrador es la de gestionar los usuarios del sistema. La aplicación recomendada en las distribuciones *Debian* y derivadas como *Ubuntu* para crear un usuario es `adduser`, a la que se le pasa como parámetro el *nombre de usuario*.

Es un programa interactivo que espera se le ingresen datos asociados con el usuario que se está creando: la contraseña de ese usuario, el nombre completo, el número de la habitación u oficina, el teléfono de la oficina, el teléfono de la casa y otro dato adicional que se quiera incorporar al directorio de usuarios.

El programa `adduser` utiliza para su configuración el archivo `/etc/adduser.conf` en el cual se pueden definir todos los parámetros globales de la aplicación.

Ofrece una interfaz más sencilla para programas de bajo nivel como `useradd`, `groupadd` y `usermod`, seleccionando valores para el identificador de usuario (**UID**) e identificador de grupo de usuarios (**GID**) conforme a las normas de Debian.

Como es un comando de administración, debe ejecutarse con permisos elevados, por lo que hay que anteponer el comando `sudo` al comando de creación de usuario.

Si se invoca con un argumento correspondiente al nombre de usuario y no tiene opciones, añadirá un usuario normal. Para este usuarios elegirá el primer **UID** disponible dentro del rango especificado para usuarios normales en el fichero de configuración, aunque se podría elegir uno manualmente usando la opción `--uid`.

Para crear el usuario `lmartinez`, se ejecuta el comando:

```
$ sudo adduser lmartinez
Añadiendo el usuario 'lmartinez' ...
Añadiendo el nuevo grupo 'lmartinez' (1001) ...
Añadiendo el nuevo usuario 'lmartinez' (1001) con grupo 'lmartinez' ...
Creando el directorio personal '/home/lmartinez' ...
Copiando los ficheros desde '/etc/skel' ...
Introduzca la nueva contraseña de UNIX:
Vuelva a escribir la nueva contraseña de UNIX:
passwd: contraseña actualizada correctamente
Cambiando la información de usuario para lmartinez
Introduzca el nuevo valor, o presione INTRO para el predeterminado
  Nombre completo []: Leonardo Martinez
  Número de habitación []:
  Teléfono del trabajo []:
  Teléfono de casa []:
  Otro []: Docente
¿Es correcta la información? [S/n] s
$
```

En el proceso se crea el directorio personal del usuario, donde se copia el contenido del directorio `/etc/skel` que provee al usuario un conjunto de directorios y archivos de configuración estándar.

El contenido del directorio `/etc/skel` es la plantilla para un usuario normal del sistema y se puede adaptar a las necesidades de uso del equipo que se está configurando. Es esencial tener en cuenta que el contenido de la plantilla se copia en el directorio personal del usuario en el momento de ser creado, si se modifica este directorio, sólo será efectivo para los usuarios creados con posterioridad a los cambios, por lo tanto, los usuarios existentes se deben actualizar manualmente.

Se pueden utilizar opciones en la ejecución del comando para evitar el ingreso interactivo de la información. Esto es muy útil cuando se necesita crear usuarios dentro de un proceso automatizado con un *script de shell*.

El comando siguiente tiene el mismo efecto que el anterior pero evitando ingresar la información en forma interactiva:

```
$ sudo adduser --disabled-password --gecos "Linus Torvalds,,,Alumno" ltorvalds

Añadiendo el usuario 'ltorvalds' ...
Añadiendo el nuevo grupo 'ltorvalds' (1003) ...
Añadiendo el nuevo usuario 'ltorvalds' (1003) con grupo 'ltorvalds' ...
Creando el directorio personal '/home/ltorvalds' ...
Copiando los ficheros desde '/etc/skel' ...
```

Se verifica que se haya incorporado correctamente en la base de usuarios:

```
$ tail -3 /etc/passwd

lmartinez:x:1001:1001:Leonardo Martinez,,,Docente:/home/lmartinez:/bin/bash
rstallman:x:1002:1002:Richard Stallman,,,Alumno:/home/rstallman:/bin/bash
ltorvalds:x:1003:1003:Linus Torvalds,,,Alumno:/home/ltorvalds:/bin/bash
```


Con el comando `chpasswd` se asigna la contraseña al usuario. Posteriormente éste podrá cambiarla con el comando `passwd` una vez iniciada la sesión.

```
$ sudo echo "ltorvalds:penguin" | sudo chpasswd
```

1.2 Bases de datos de grupos

Un grupo *GNU/Linux* es una entidad que incluye varios usuarios del sistema para que puedan compartir archivos fácilmente utilizando el sistema de permisos. También se puede restringir el uso de ciertos programas a un grupo específico. Cada usuario puede ser miembro de varios grupos, uno de los cuales es su *grupo principal* que se crea de forma predeterminada durante la creación del usuario.

Cada archivo que cree el usuario pertenece a él así como también a su grupo principal. Esto no es siempre el comportamiento deseado; por ejemplo, cuando el usuario necesita trabajar en un directorio compartido por un grupo distinto a su grupo principal. Esta necesidad se resuelve creando grupos relacionados a las actividades que realizan los usuarios e incorporando a estos en cada grupo de acuerdo al rol que cumplen. Por ejemplo, en una empresa podríamos tener grupos como *administración*, *ventas* y *personal*.

El archivo de configuración está en `/etc/group`, también en texto plano. El archivo `/etc/group` de la máquina virtual que se utiliza de ejemplo es:

```
$ cat /etc/group
root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:syslog,administrador
tty:x:5:
disk:x:6:
lp:x:7:
mail:x:8:
news:x:9:
uucp:x:10:
man:x:12:
proxy:x:13:
kmem:x:15:
dialout:x:20:
fax:x:21:
voice:x:22:
cdrom:x:24:administrador
floppy:x:25:
tape:x:26:
sudo:x:27:administrador
audio:x:29:pulse
dip:x:30:administrador
www-data:x:33:
backup:x:34:
operator:x:37:
list:x:38:
irc:x:39:
src:x:40:
gnats:x:41:
```

```
shadow:x:42:
utmp:x:43:
video:x:44:
sasl:x:45:
plugdev:x:46:administrador
staff:x:50:
games:x:60:
users:x:100:
nogroup:x:65534:
systemd-journal:x:101:
systemd-timesync:x:102:
systemd-network:x:103:
systemd-resolve:x:104:
systemd-bus-proxy:x:105:
input:x:106:
crontab:x:107:
syslog:x:108:
netdev:x:109:
messagebus:x:110:
uidd:x:111:
ssl-cert:x:112:
lpadmin:x:113:administrador
lightdm:x:114:
nopasswdlogin:x:115:
whoopsie:x:116:
mlocate:x:117:
ssh:x:118:
avahi-autoipd:x:119:
avahi:x:120:
bluetooth:x:121:
scanner:x:122:saned
colord:x:123:
pulse:x:124:
pulse-access:x:125:
rtkit:x:126:
saned:x:127:
administrador:x:1000:
sambashare:x:128:administrador
```

Cada registro del archivo corresponde a un grupo del sistema y está definido como una serie de campos separados por el carácter `:` con la siguiente información:

Nombre de grupo

Es la cadena de caracteres que se utiliza como identificación del grupo.

Contraseña

El uso es opcional, sólo es utilizada para unirse a un grupo cuando no es un miembro con el comando `newgrp`.

GID (Group Identifier number)

El número de identificador de grupo.

Lista de miembros

Indica los usuarios que pertenecen al grupo. El listado consiste en los *nombres de usuario* separados por coma.

1.2.1 Creación de grupos

La aplicación recomendada en las distribuciones *Debian* y derivadas como *Ubuntu* para crear un usuario es `addgroup`, a la que se le pasa como parámetro el *nombre del nuevo grupo*. En realidad este comando es un enlace simbólico al comando `adduser` que al ser ejecutado de esa manera crea un nuevo grupo en lugar de un usuario. El mismo resultado se obtendría ejecutando `adduser --group <nombre_grupo>`

Si se invoca con un argumento correspondiente al nombre de grupo y no tiene opciones, añadirá un grupo normal. Para este grupo elegirá el primer **GID** disponible dentro del rango especificado para grupos normales en el fichero de configuración, aunque se podría elegir uno manualmente usando la opción `--gid`. El grupo se crea sin usuarios, por lo tanto, habrá que incorporárselos.

Para crear el grupo `ventas`, se ejecuta el comando:

```
$ sudo addgroup ventas
Añadiendo el grupo `ventas' (GID 1004) ...
Hecho.

$ tail -4 /etc/group
lmartinez:x:1001:
rstallman:x:1002:
ltorvalds:x:1003:
ventas:x:1004:
$
```

1.2.2 Agregar usuarios a grupos

Para incorporar los usuarios a los grupos existentes, se utiliza el comando `adduser` con dos parámetros, el primero es el *nombre de usuario* y el segundo es el *grupo* al que se quiere agregarlo.

Para agregar el usuario `ltorvalds` al grupo `ventas`, se ejecuta el comando:

```
$ sudo adduser ltorvalds ventas
Añadiendo al usuario `ltorvalds' al grupo `ventas' ...
Añadiendo al usuario ltorvalds al grupo ventas
Hecho.

$ tail -4 /etc/group
lmartinez:x:1001:
rstallman:x:1002:
ltorvalds:x:1003:
ventas:x:1004:ltorvalds
$
```

1.2.3 Eliminación de usuarios y grupos

Los comandos `deluser` y `delgroup` se utilizan para eliminar usuarios y grupos del sistema respectivamente.

Al igual que los comandos de creación, los programas tienen un archivo de configuración, en este caso en `/etc/deluser.conf` y `/etc/adduser.conf`. Proporcionan una interfaz más sencilla para los programas `userdel` y `groupdel`, eliminando opcionalmente el directorio personal o incluso todos los ficheros del sistema pertenecientes al usuario, ejecutar un script personalizado, y otras características.

Como son comandos de administración, deben ejecutarse con permisos elevados, por lo que hay que anteponer el comando `sudo`.

Para eliminar un usuario el comando `deluser` debe ejecutarse con el *nombre del usuario* como parámetro.

```
$ tail -3 /etc/passwd

lmartinez:x:1001:1001:Leonardo Martinez,,,:/home/lmartinez:/bin/bash
rstallman:x:1002:1002:Richard Stallman,,,:/home/rstallman:/bin/bash
ltorvalds:x:1003:1003:Linus Torvalds,,,:/home/ltorvalds:/bin/bash
$

$ sudo deluser rstallman
Eliminando al usuario `rstallman' ...
Aviso: el grupo `rstallman' no tiene más miembros.
Hecho.
$

$ tail -3 /etc/passwd
sshd:x:121:65534::/var/run/sshd:/usr/sbin/nologin
lmartinez:x:1001:1001:Leonardo Martinez,,,:/home/lmartinez:/bin/bash
ltorvalds:x:1003:1003:Linus Torvalds,,,:/home/ltorvalds:/bin/bash
$
```

También elimina el *grupo principal* del usuario.

```
~$ tail -4 /etc/group
sambashare:x:128:administrador
lmartinez:x:1001:
ltorvalds:x:1003:
ventas:x:1004:ltorvalds
$
```

Pero al verificar el directorio `/home` se observa que el directorio personal del usuario eliminado todavía existe, por lo que deberá eliminarse manualmente.

```
$ ls -la /home/
total 40
drwxr-xr-x  7 root          root          4096 ago 16 00:52 .
drwxr-xr-x 24 root          root          4096 ago 14 19:42 ..
drwxr-xr-x 16 administrador administrador 4096 ago 17 08:40 administrador
drwxr-xr-x  3 lmartinez     lmartinez     4096 ago 15 22:46 lmartinez
drwx-----  2 root          root          16384 may 11 23:46 lost+found
drwxr-xr-x  3 ltorvalds     ltorvalds     4096 ago 16 00:52 ltorvalds
drwxr-xr-x  3              1002              1002 4096 ago 16 00:44 rstallman

$ sudo rm -R /home/rstallman/
```

```
$ ls -la /home/
total 36
drwxr-xr-x  6 root          root          4096 ago 17 18:25 .
drwxr-xr-x 24 root          root          4096 ago 14 19:42 ..
drwxr-xr-x 16 administrador administrador 4096 ago 17 08:40 administrador
drwxr-xr-x  3 lmartinez     lmartinez     4096 ago 15 22:46 lmartinez
drwx----- 2 root          root          16384 may 11 23:46 lost+found
drwxr-xr-x  3 ltorvalds     ltorvalds     4096 ago 16 00:52 ltorvalds
$
```

Esto se debe a que por omisión, el comando `deluser` eliminará el usuario, pero no su directorio personal ni su directorio de correo o cualquier otro archivo del sistema que le pertenezca. Para eliminar el directorio personal y de correo se debe usar la opción `--remove-home` para borrar el directorio **\$HOME** del usuario y, si se pretende borrar todos los archivos pertenecientes al usuario en el sistema el parámetro a utilizar es `--remove-all-files` que incluye el directorio **\$HOME**.

Al igual que `addgroup`, el comando `delgroup` es un enlace simbólico al comando `deluser` que al ser ejecutado de esa manera elimina un grupo en lugar de un usuario. El mismo resultado se obtendría ejecutando `deluser --group <nombre_grupo>`.

No se puede eliminar el *grupo principal* de un usuario existente, en cambio el resto de los grupos se pueden eliminar sin problemas aún teniendo usuarios asociados. Si se pretende controlar que el grupo esté vacío para poder eliminarlo se usa el argumento o parámetro `--only-if-empty`.

Para borrar el grupo `ventas`, que tiene asociado el usuario `ltorvalds`, es suficiente con ejecutar el comando `sudo delgroup ventas`:

```
$ tail -4 /etc/group
ltorvalds:x:1003:
ventas:x:1004:ltorvalds
alovelace:x:1002:
personal:x:1005:alovelace

$ sudo delgroup ventas
Eliminando al grupo `ventas' ...
Hecho.

$ tail -4 /etc/group
lmartinez:x:1001:
ltorvalds:x:1003:
alovelace:x:1002:
personal:x:1005:alovelace
$
```

Para borrar el grupo `personal` siempre y cuando no tenga usuarios asociados (tiene asociado el usuario `alovelace`), es necesario con ejecutar el comando con la opción `--only-if-empty`:

```
$ sudo delgroup --only-if-empty personal
/usr/sbin/delgroup: ¡El grupo `personal' no está vacío!
$
```

2 Más herramientas de gestión de usuarios y grupos

Además de los comandos de creación y eliminación de usuarios y grupos hay otras herramientas que se utilizan para gestionar.

2.1 El comando `id`

El programa `id` muestra la identidad de un usuario junto con la lista de grupos a los que pertenece. Debido a que el acceso a algunos archivos o dispositivos puede estar limitados a miembros de ciertos grupos, puede ser útil verificar a qué grupos se pertenece.

Si no se indica el *nombre de usuario* como parámetro, el comando muestra la información del usuario de la sesión.

```
$ id
uid=1000(administrador) gid=1000(administrador) grupos=1000(administrador),4(adm)
,24(cdrom),27(sudo),30(dip),46(plugdev),113(lpadmin),128(sambashare)

$ id lmartinez
uid=1001(lmartinez) gid=1001(lmartinez) grupos=1001(lmartinez)

$ id alovelace
uid=1002(alovelace) gid=1002(alovelace) grupos=1002(alovelace),1005(personal)

$
```

Este comando tiene algunas opciones interesantes como `-u` que muestra el **UID**, `-g` que indica el **GID** del grupo principal del usuario y `-G` que lista los **GID** de los grupos a los que pertenece el usuario. Si a estas opciones le agregamos el parámetro `-n` muestra el nombre del usuario o del grupo en lugar del **UID** o del **GID**. Estas opciones son muy interesante para su uso en *scripts de shell*.

Para el usuarios en sesión se ejecuta el comando sin hacer referencia al mismo:

```
$ id -u
1000

$ id -g
1000

$ id -G
1000 4 24 27 30 46 113 128

$ id -un
administrador

$ id -gn
administrador

$ id -Gn
administrador adm cdrom sudo dip plugdev lpadmin sambashare
```

Para el usuario `alovelace` se tiene:

```
$ id -u avelace
1002

$ id -g avelace
1002

$ id -G avelace
1002 1005

$ id -un avelace
avelace

$ id -gn avelace
avelace

$ id -Gn avelace
avelace personal
```

2.2 El comando `newusers`

Este comando es particularmente útil cuando necesitamos modificar o crear varios usuarios al mismo tiempo. El comando lee un archivo de texto plano y utiliza la información para actualizar los usuarios existentes o crear nuevos. Cada línea del archivo de texto es un registro con el formato de la base de usuarios `/etc/passwd`:

```
pw_name:pw_passwd:pw_uid:pw_gid:pw_gecos:pw_dir:pw_shell
```

Donde:

- **pw_name** es el nombre de usuario.
- **pw_passwd** contiene una cadena de texto que representa la contraseña plana. Por esta razón el archivo debe tener los permisos correspondientes para proteger la información.
- **pw_uid** es el **UID** asignado al usuario. Si está vacío se utiliza el disponible de acuerdo a la configuración del sistema.
- **pw_gid** es el **GID** asignado al usuario. Si está vacío se utiliza el disponible de acuerdo a la configuración del sistema.
- **pw_gecos** es la información del usuario.
- **pw_dir** es el directorio **\$HOME** del usuario.
- **pw_shell** es el intérprete de comandos asignado al usuario.

Como ejemplo se tiene el archivo `usuarios_nuevos.txt` con el siguiente contenido:

```
$ cat usuarios_nuevos.txt
aturing:aimachine:::Alan Turing,,,Alumno:/home/aturing:/bin/bash
aswartz:jstor2010:::Aaron Swartz,,,Alumno:/home/aswartz:/bin/bash
```

Se ajustan los permisos del archivo por la información sensible:

```
$ chmod 700 usuarios_nuevos.txt

$ ls -la usuarios_nuevos.txt
-rwx----- 1 administrador administrador 133 ago 17 23:34 usuarios_nuevos.txt
```

Se ejecuta el comando `newusers` para incorporar los nuevos usuarios:

```
$ sudo newusers usuarios_nuevos.txt

$ tail -5 /etc/passwd
lmartinez:x:1001:1001:Leonardo Martinez,,,:/home/lmartinez:/bin/bash
ltorvalds:x:1003:1003:Linus Torvalds,,,:/home/ltorvalds:/bin/bash
alovelace:x:1002:1002:Ada Lovelace,,,:/home/alovelace:/bin/bash
aturing:x:1004:1004:Alan Turing,,,:/home/aturing:/bin/bash
aswartz:x:1005:1006:Aaron Swartz,,,:/home/aswartz:/bin/bash

$ tail -6 /etc/group
lmartinez:x:1001:
ltorvalds:x:1003:
alovelace:x:1002:
personal:x:1005:alovelace
aturing:x:1004:
aswartz:x:1006:

$ ls -l /home/ | grep -v lost+found
total 40
drwxr-xr-x 16 administrador administrador 4096 ago 17 23:34 administrador
drwxr-xr-x  3 alovelace      alovelace      4096 ago 17 18:31 alovelace
drwxr-xr-x  2 aswartz       aswartz        4096 ago 17 23:55 aswartz
drwxr-xr-x  2 aturing       aturing        4096 ago 17 23:55 aturing
drwxr-xr-x  3 lmartinez     lmartinez     4096 ago 15 22:46 lmartinez
drwxr-xr-x  3 ltorvalds     ltorvalds     4096 ago 16 00:52 ltorvalds
```

2.3 El comando `passwd`

Este comando tiene muchos usos interesantes para gestionar las contraseñas de los usuarios. Pero la funcionalidad más importante es la de cambiar la contraseña de usuario.

Si se pretende cambiar la contraseña del usuario de sesión, el comando `passwd` debe ejecutarse sin argumentos. Solicitará la contraseña actual y luego pedirá ingresar la nueva dos veces para corroborar consistencia.

Algunos de los parámetros interesantes son:

- **-d, --delete:** Borra la contraseña del usuario, la hace vacía. Es una manera rápida de deshabilitar una contraseña para una cuenta y la deja sin contraseña.
- **-l, --lock:** Bloquea la contraseña del usuario, le agrega el caracter `!` al inicio del *hash* de la contraseña haciendo imposible que el cifrado de una cadena de texto coincida. De esta manera se deshabilita la cuenta.
- **-u, --unlock:** Desbloquea la contraseña del usuario. Esta opción rehabilita la cuenta dejando la contraseña como estaba antes de agregarle el caracter `!` con la opción `-l`.

3 Bibliografía

- Páginas de Manual (Man Pages) del sistema operativo.

La bibliografía que se indica a continuación corresponde a material que provee una visión interesante sobre los temas propuestos en esta unidad. Es documentación más completa e incluso más extensiva en el desarrollo de algunos temas.

- The Debian Administrator's Handbook, Raphaël Hertzog and Roland Mas, (<https://debian-handbook.info/>)
- Administración de sistemas GNU/Linux, Máster universitario en Software Libre, Universitat Oberta de Catalunya
- Básicamente GNU/Linux, Antonio Perpiñan, Fundación Código Libre Dominicano (<http://www.codigolibre.org>)