



Instituto Tecnológico Las Américas  
(ITLA)

Sistemas Operativos 3 (SO3)

Daniel Alejandro Moreno Martínez

Matrícula: 2010-2946



## How to

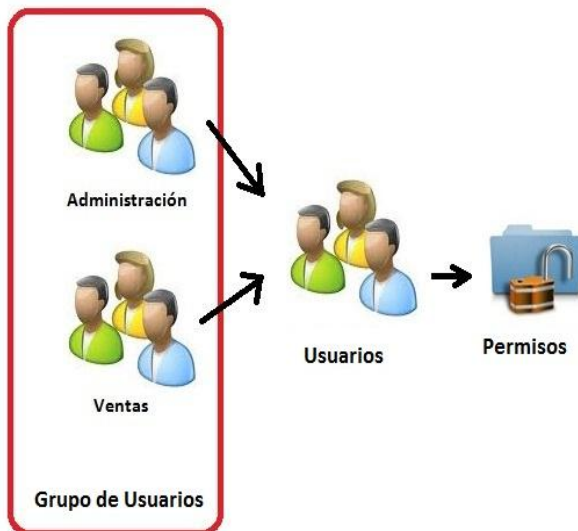
!!! How to: Creación de usuarios, grupos y asignación de permisos !!!

### Creación de Usuarios, Grupos y Asignación de Permisos

A continuación en la siguiente práctica procederemos a crear cuentas de usuarios, grupos de usuarios y además asignarle permisos.

Un **usuario** es un individuo que utiliza una computadora, sistema operativo, servicio o cualquier sistema, por lo tanto no podemos asignarles los mismos permisos a todos los usuarios, eso es algo que vamos a aprender aquí.

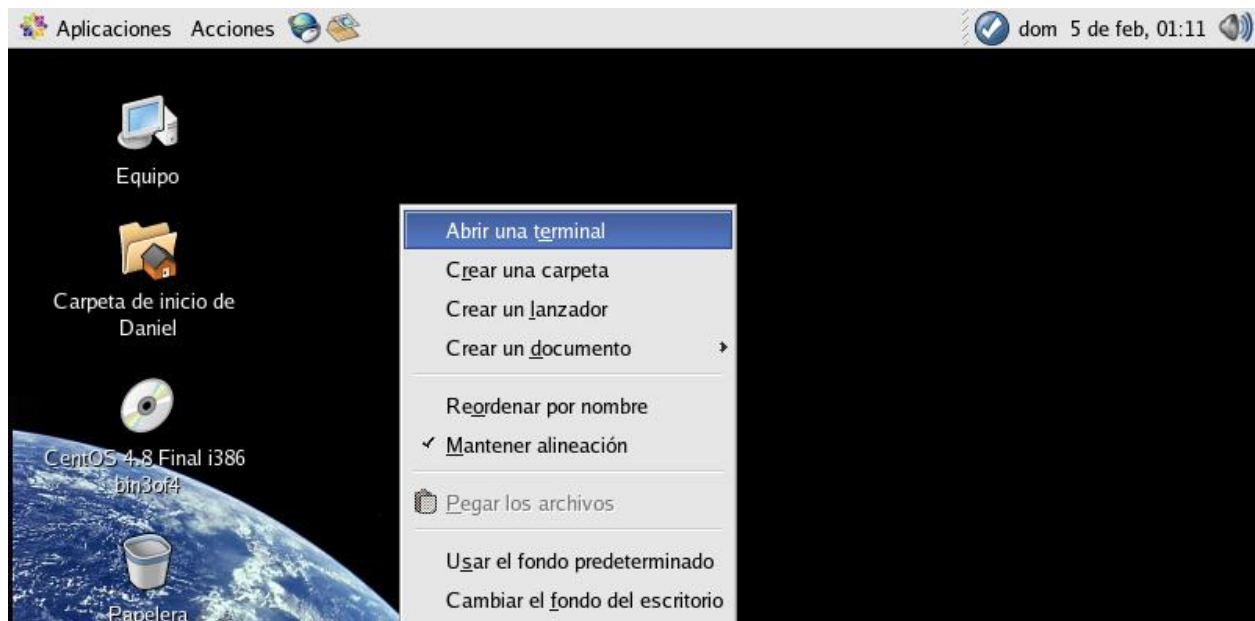
La práctica que vamos a realizar a continuación está orientada al sistema operativo **CentOS (Distribución basada en Linux)**.



Para crear un usuario se debe modificar el archivo **/etc/passwd**. Podemos hacerlo de forma segura utilizando el comando de linux **vipw** (sirve para bloquear el archivo mientras lo estás modificando de forma que nadie más pueda editarlo).

## Creación de Usuarios

Lo primero que debemos hacer es abrir una terminal, esto lo hacemos dando click derecho en el escritorio.

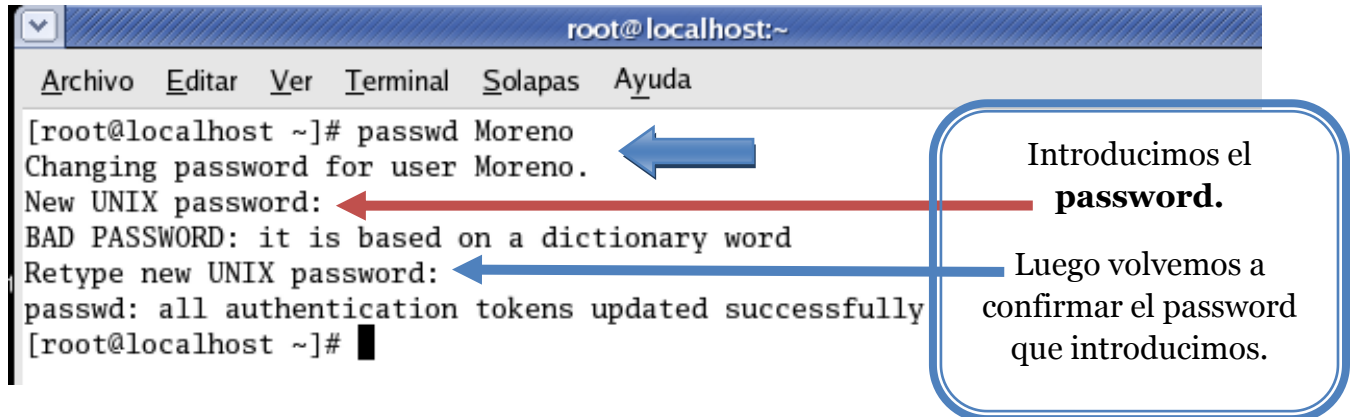


Luego nos aparecerá la terminal donde procederemos a introducir los comandos a utilizar. El primer comando que utilizaremos es el **comando Su** – para pasar al **modo root** > luego introducimos el **password del usuario root**.

Para crear un usuario, debemos pasar a modo **root** y utilizamos la línea de comando **useradd** para crear un usuario, en este caso llamado **Daniel**, y le pondremos un **UID** manualmente. *En sistemas tipo Unix*, los usuarios son representados por un identificador de usuario, normalmente abreviado como **UID**. Las características básicas son: \* El rango de los valores de los **UID** varía entre los diferentes sistemas. \*\* *Como mínimo los UID* deben estar comprendidos entre **0 y 32,767**. En este caso el que vamos a utilizar es **1000**.

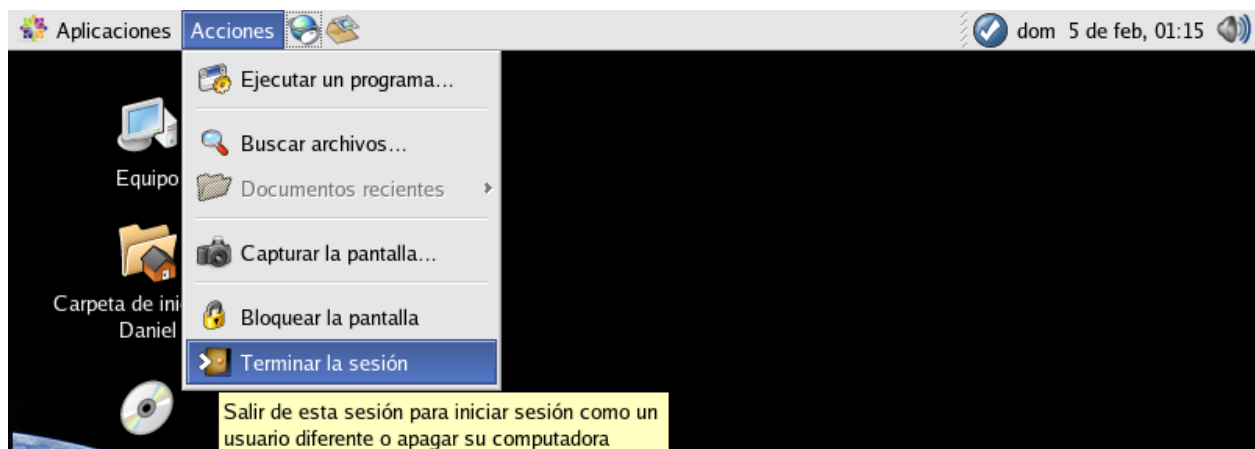
```
root@localhost:~
Archivo  Editar  Ver  Terminal  Solapas  Ayuda
[Daniel@localhost ~]$ su -
Password:
[root@localhost ~]# useradd Moreno -u 1000
[root@localhost ~]# █
```

Para definir la contraseña de este usuario utilizamos el comando **passwd**, seguido del **nombre de usuario** al que le vamos a definir dicho password.

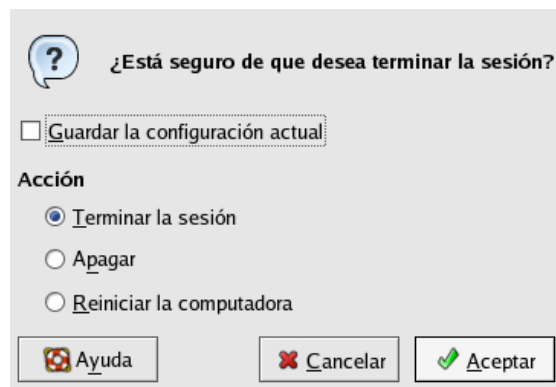


Una vez creado el nuevo usuario, podemos terminar la sesión como **root** y **logearnos** como **NelsonMosquea**. *Lo hacemos de la siguiente forma.*

Damos click en Acciones y luego en Terminar sesión, Luego nos pide que introduzcamos **el nuevo usuario** creado y por **último la password** que definimos para este.



Aquí Seleccionamos **Terminar la sesión** y hacemos clic en **Aceptar** para logearnos en el nuevo usuario.



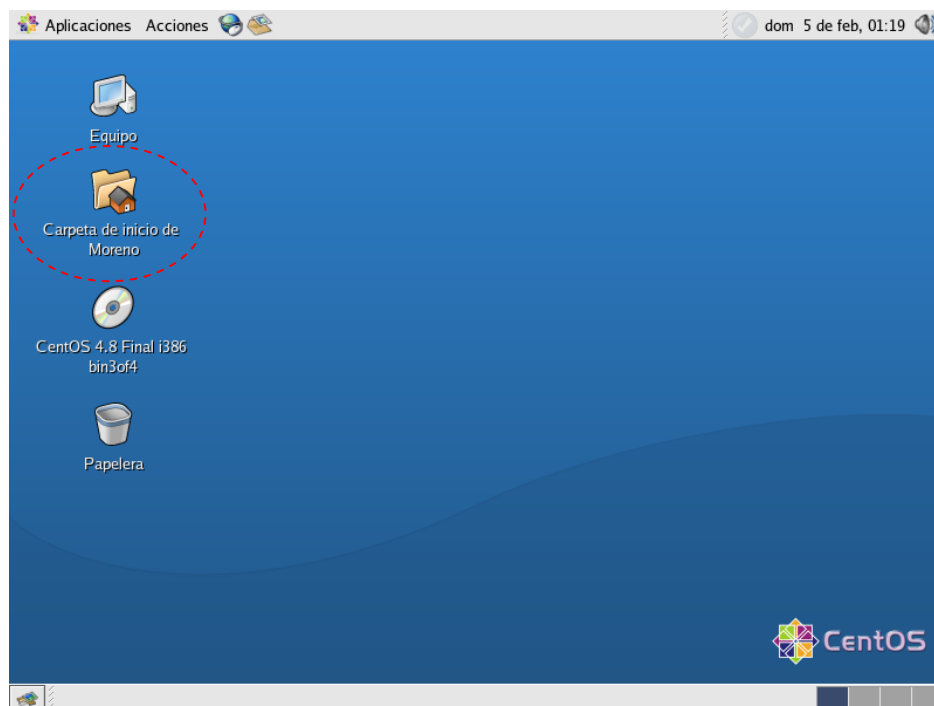
Aquí procedemos a logearnos con el usuario que acabamos de crear. Introducimos el nombre y la contraseña o password del usuario.



Esperamos mientras carga la configuración del usuario.



Luego de esto podremos ver cómo hemos iniciado sesión con **nuestro nuevo usuario**.



Dentro de nuestro usuario "**Moreno**", podemos ver nuestro directorio con el comando **pwd** (En este caso, **/home/Moreno**). Para ver que no tenemos permisos para acceder a las home de otros usuarios, podemos intentar cambiar de directorio para entrar en algún home. Al ejecutar, por ejemplo, **cd /home/Daniel**, nos dará un error indicando que no tenemos **los permisos**.

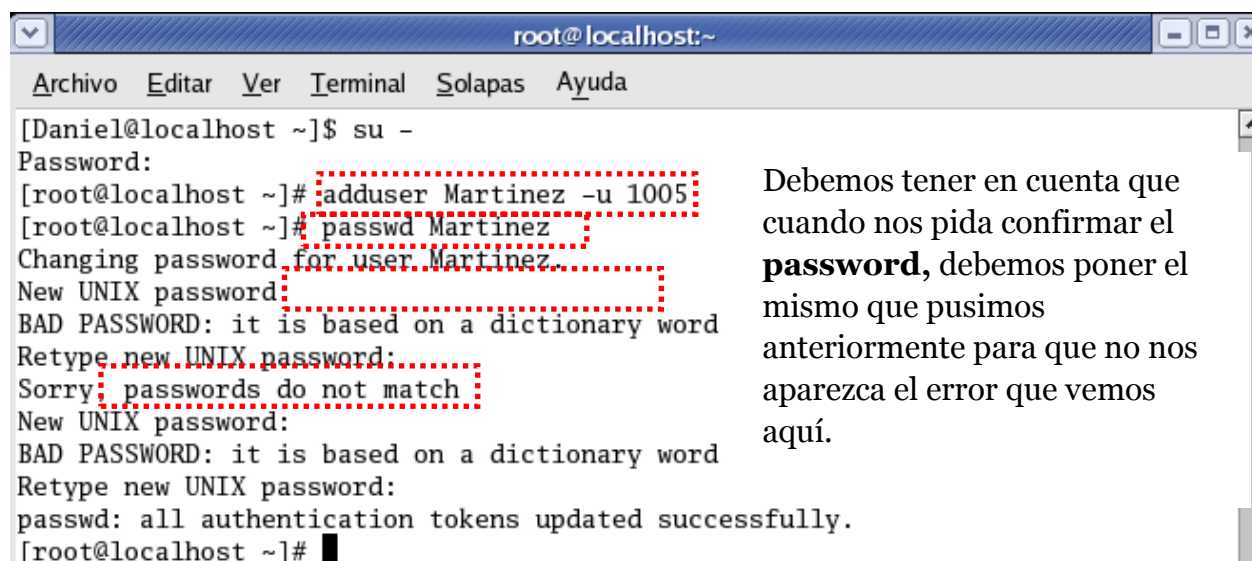
```
Moreno@localhost:~  
Archivo Editar Ver Terminal Solapas Ayuda  
[Moreno@localhost ~]$ pwd  
/home/Moreno  
[Moreno@localhost ~]$ cd /home/Daniel  
bash: cd: /home/Daniel: Permiso denegado  
[Moreno@localhost ~]$
```

Ahora vamos a volver al usuario root, para comprobar que nuestro usuario está añadido en **/etc/passwd**. Para esto, podemos usar la orden **more /etc/passwd**.

```
Moreno@localhost:~  
Archivo Editar Ver Terminal Solapas Ayuda  
[Moreno@localhost ~]$ more /etc/passwd  
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin  
nobody:x:99:99:Nobody:/:/sbin/nologin  
dbus:x:81:81:System message bus:/:/sbin/nologin  
vcsa:x:69:69:virtual console memory owner:/dev:/sbin/nologin  
rpm:x:37:37:/:/var/lib/rpm:/sbin/nologin  
haldaemon:x:68:68:HAL daemon:/:/sbin/nologin  
netdump:x:34:34:Network Crash Dump user:/var/crash:/bin/bash  
nscd:x:28:28:NSCD Daemon:/:/sbin/nologin  
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin  
rpc:x:32:32:Portmapper RPC user:/:/sbin/nologin  
mailnull:x:47:47:/:/var/spool/mqueue:/sbin/nologin  
smmsp:x:51:51:/:/var/spool/mqueue:/sbin/nologin  
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin  
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin  
pcap:x:77:77:/:/var/arpwatch:/sbin/nologin  
apache:x:48:48:Apache:/var/www:/sbin/nologin  
squid:x:23:23:/:/var/spool/squid:/sbin/nologin  
webalizer:x:67:67:Webalizer:/var/www/usage:/sbin/nologin  
xfs:x:43:43:X Font Server:/etc/X11/fs:/sbin/nologin  
ntp:x:38:38:/:/etc/ntp:/sbin/nologin  
gdm:x:42:42:/:/var/gdm:/sbin/nologin  
Daniel:x:500:500:Daniel Moreno:/home/Daniel:/bin/bash  
Moreno:x:1000:1000:/:/home/Moreno:/bin/bash  
[Moreno@localhost ~]$
```

Al final del archivo aparece nuestro usuario **Moreno**, con el **UID 1000**. Seguramente se habrán fijado que en el campo de contraseña, aparece **una x**. Esto es porque por defecto se instala un programa llamado **shadow** que oculta las contraseñas cifradas en otro archivo (**/etc/shadow**).

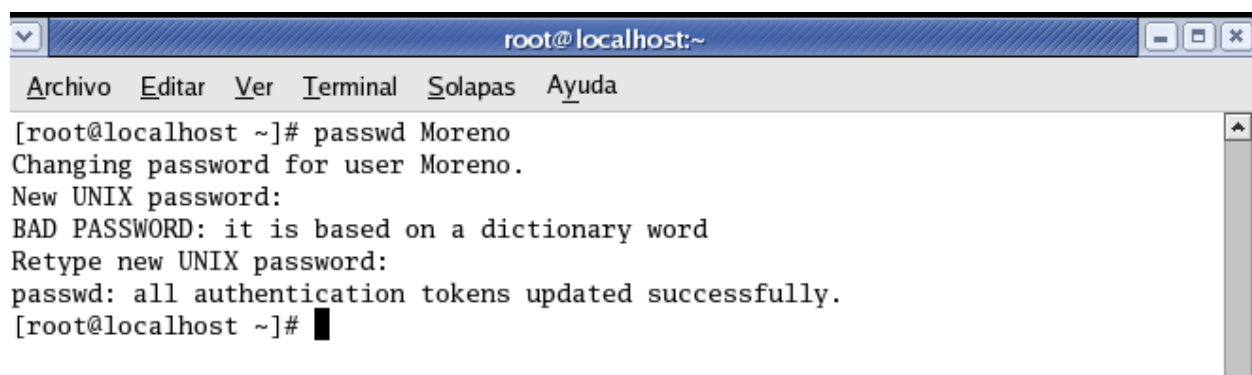
También tenemos el comando **adduser**, con el cual también podemos agregar usuarios.



```
root@localhost:~  
Archivo Editar Ver Terminal Solapas Ayuda  
[Daniel@localhost ~]$ su -  
Password:  
[root@localhost ~]# adduser Martinez -u 1005  
[root@localhost ~]# passwd Martinez  
Changing password for user Martinez.  
New UNIX password:  
BAD PASSWORD: it is based on a dictionary word  
Retype new UNIX password:  
Sorry, passwords do not match  
New UNIX password:  
BAD PASSWORD: it is based on a dictionary word  
Retype new UNIX password:  
passwd: all authentication tokens updated successfully.  
[root@localhost ~]#
```

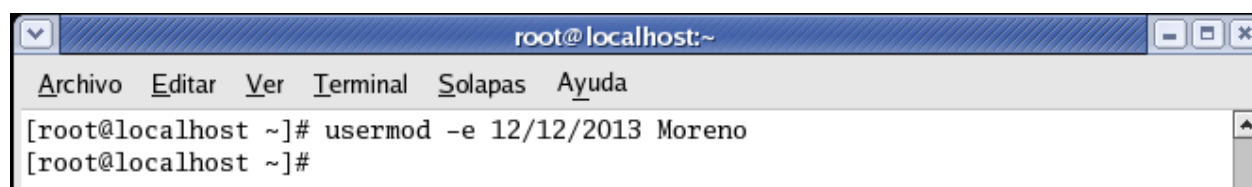
Debemos tener en cuenta que cuando nos pida confirmar el **password**, debemos poner el mismo que pusimos anteriormente para que no nos aparezca el error que vemos aquí.

Para cambiar el **password** a un usuario escribimos **passwd nombre de usuario** y nos pedirá la nueva **passwd**.



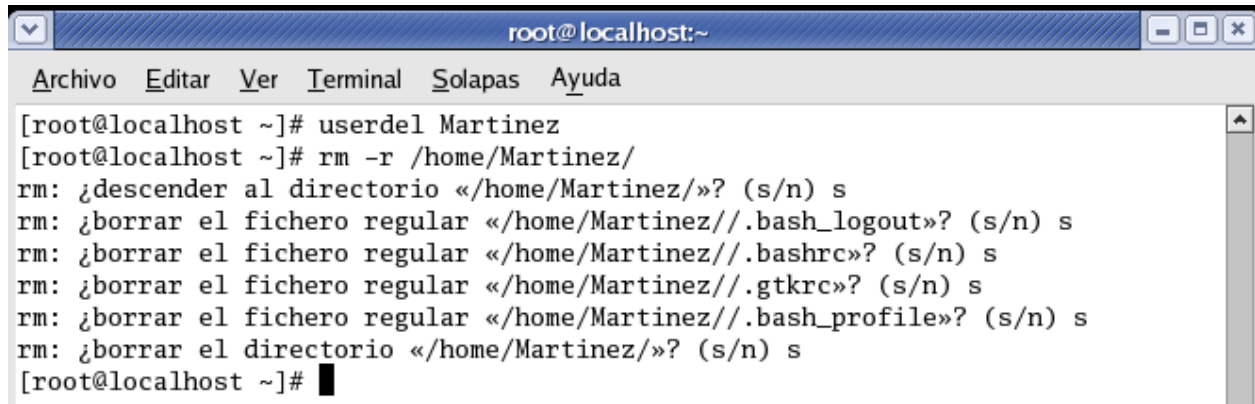
```
root@localhost:~  
Archivo Editar Ver Terminal Solapas Ayuda  
[root@localhost ~]# passwd Moreno  
Changing password for user Moreno.  
New UNIX password:  
BAD PASSWORD: it is based on a dictionary word  
Retype new UNIX password:  
passwd: all authentication tokens updated successfully.  
[root@localhost ~]#
```

Para expirar la **passwd** de un usuario escribimos **usermod -e 05/01/2012** que en este caso es la fecha de expiración y **el nombre del usuario**.



```
root@localhost:~  
Archivo Editar Ver Terminal Solapas Ayuda  
[root@localhost ~]# usermod -e 12/12/2013 Moreno  
[root@localhost ~]#
```

Para eliminar un usuario utilizamos el comando **userdel** y para el eliminar el **directorio home del usuario** utilizamos **rm -r /home/ (nombre de usuario)**. El programa nos pregunta también si queremos eliminar los archivos de configuración dentro del home.



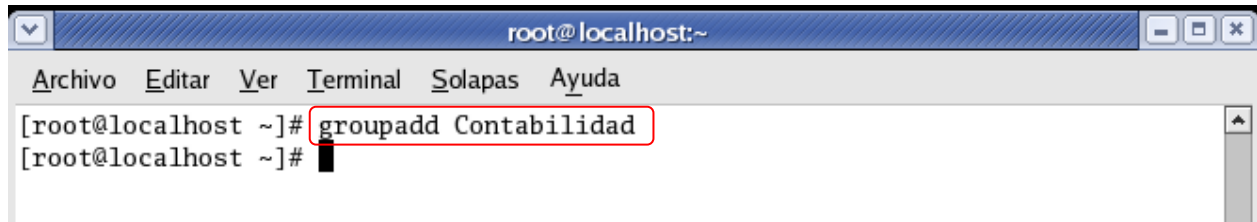
```
root@localhost:~
Archivo Editar Ver Terminal Solapas Ayuda
[root@localhost ~]# userdel Martinez
[root@localhost ~]# rm -r /home/Martinez/
rm: ¿descender al directorio «/home/Martinez/»? (s/n) s
rm: ¿borrar el fichero regular «/home/Martinez//.bash_logout»? (s/n) s
rm: ¿borrar el fichero regular «/home/Martinez//.bashrc»? (s/n) s
rm: ¿borrar el fichero regular «/home/Martinez//.gtkrc»? (s/n) s
rm: ¿borrar el fichero regular «/home/Martinez//.bash_profile»? (s/n) s
rm: ¿borrar el directorio «/home/Martinez/»? (s/n) s
[root@localhost ~]#
```

De este forma nuestro usuario ha sido eliminado totalmente de nuestro sistema.

## Creación de grupos de usuario

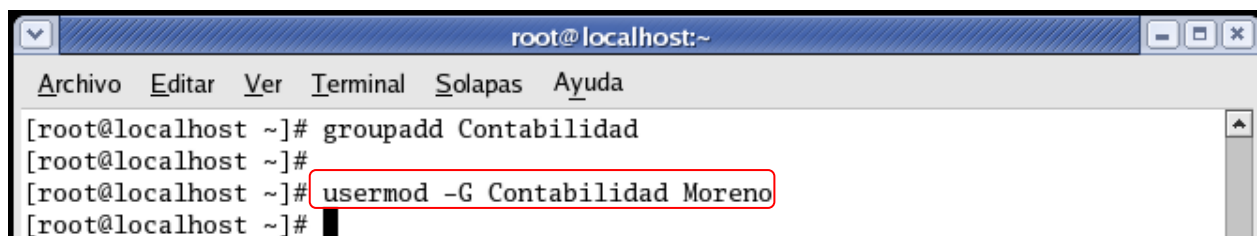
A continuación procederemos a crear grupos con el comando **groupadd** y eliminarlos con el comando **groupdel**.

Por ejemplo, podemos crear el grupo **Contabilidad** con el comando **groupadd Contabilidad**.



```
root@localhost:~  
Archivo Editar Ver Terminal Solapas Ayuda  
[root@localhost ~]# groupadd Contabilidad  
[root@localhost ~]#
```

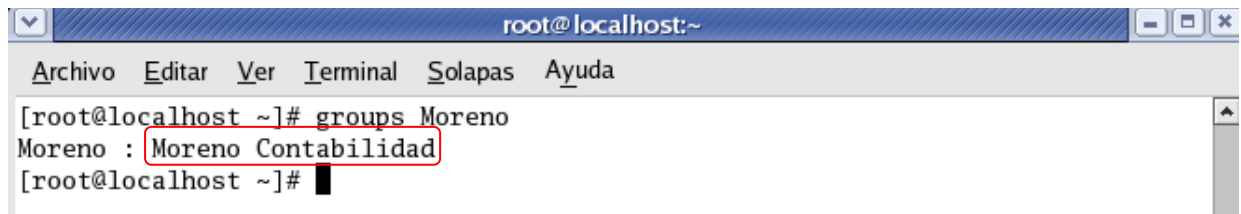
Si queremos agregar el **usuario Moreno** a este grupo utilizamos el siguiente comando.



```
root@localhost:~  
Archivo Editar Ver Terminal Solapas Ayuda  
[root@localhost ~]# groupadd Contabilidad  
[root@localhost ~]#  
[root@localhost ~]# usermod -G Contabilidad Moreno  
[root@localhost ~]#
```

Con esa orden, añadimos al **usuario Moreno** al grupo **Contabilidad**. Si nos dirigimos de nuevo al **archivo /etc/group**, veremos el grupo **Contabilidad**, conteniendo por ahora al usuario **Moreno**.

Para comprobar escribimos **groups** y el **nombre de usuario** y nos dirá los grupos a los que pertenece el usuario.



```
root@localhost:~  
Archivo Editar Ver Terminal Solapas Ayuda  
[root@localhost ~]# groups Moreno  
Moreno : Moreno Contabilidad  
[root@localhost ~]#
```



## Asignación de permisos



El sistema **de permisos en Linux** se basa en un esquema de **usuarios/grupos** que lo convierte en la base principal de la seguridad en Linux, a estos usuarios y grupos se les asignan distintos derechos sobre los **archivos y directorios**.

Esta es una de las características que ayudan a que Linux sea **casi inmune a los Virus de computadora**, los virus deben ser capaces de escribir sobre un archivo para poder infectarlo y ejecutarse de alguna manera para poder infectar más archivos, con el sistema de **permisos de Linux** los virus no pueden copiarse a cualquier archivo, si el usuario carece de permisos el virus no podrá infectar más archivos y por lo tanto **no podrá reproducirse**.

**Los permisos anteriormente mencionados son tres:**

**r: read (lectura):** Cuando el permiso de lectura está activo sobre un directorio significa que se podrá listar los recursos almacenados en él, si está asignado a un archivo se podrá leer su contenido.

**w: write (escritura):** Cuando el permiso de escritura está activo sobre un directorio significa que se podrá crear y borrar archivos en su interior, si está activado para un archivo significa que se podrá modificar su contenido.

**x: execute (ejecución):** Si el permiso de ejecución está activo sobre un directorio significa que el usuario podrá realizar otras funciones dentro de él mediante los otros permisos de lectura y escritura, y si está activo sobre un archivo se podrá ejecutarlo desde la línea de comandos.

Linux dispone de 3 comandos que permite cambiar los permisos, el propietario y el grupo de un archivo y/o directorio respectivamente:

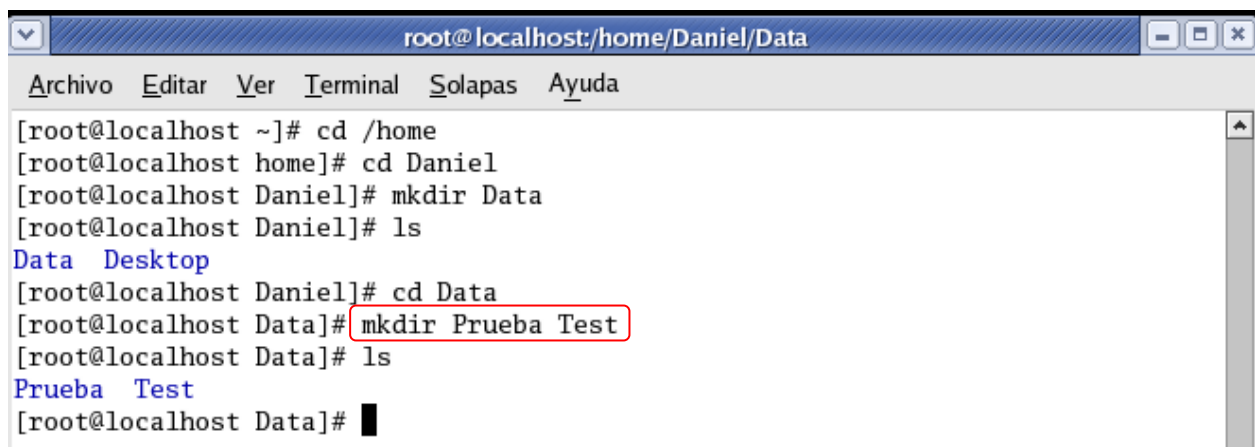
**Comando chmod:** se utiliza para cambiar los permisos del archivo o directorio.  
`$ chmod [permisos] [archivo/directorio] [opciones]`

**Comando chown:** se utiliza para cambiar el propietario del archivo o directorio.  
*# chown [nuevo usuario propietario] [archivo/directorio] [opciones]*

**Comando chgrp:** utilizado para cambiar el grupo del archivo o directorio.  
*# chgrp [nuevo grupo] [archivo/directorio] [opciones]*

	Símbolo	Descripción
Identidades	u	Es el usuario propietario del archivo o directorio
	g	Es el grupo al que pertenece el archivo o directorio
	o	Otros usuarios, el resto del mundo, ni el propietario ni su grupo
	a	Todo el mundo - propietario, grupo y otros
Permisos	r	Acceso de lectura
	w	Acceso de escritura
	x	Acceso de ejecución
Acciones	+	Añade los permisos
	-	Elimina los permisos
	=	el único permiso

Vamos a crear dos archivos los cuales utilizaremos de ejemplo al momento de explicar cómo usar **los permisos**. Los ficheros **Prueba** y **Test** serán los ejemplos que utilizaremos para aplicar permisos.

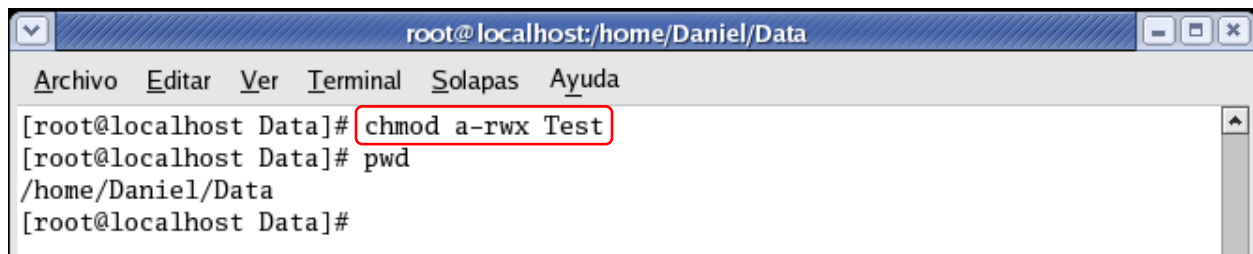


```
root@localhost:/home/Daniel/Data
Archivo  Editar  Ver  Terminal  Solapas  Ayuda
[root@localhost ~]# cd /home
[root@localhost home]# cd Daniel
[root@localhost Daniel]# mkdir Data
[root@localhost Daniel]# ls
Data Desktop
[root@localhost Daniel]# cd Data
[root@localhost Data]# mkdir Prueba Test
[root@localhost Data]# ls
Prueba Test
[root@localhost Data]#
```

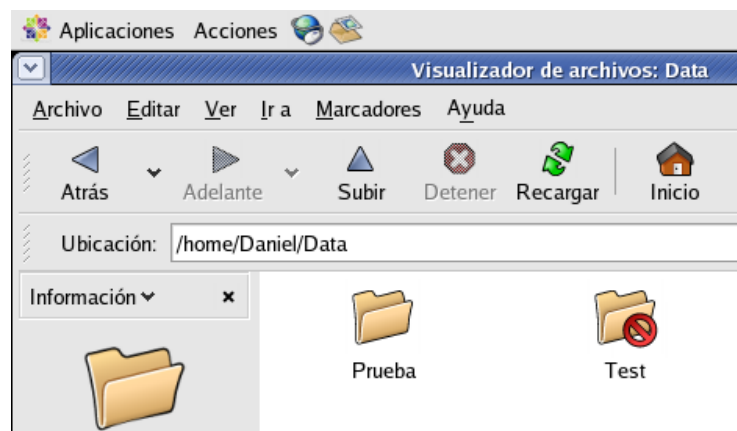
Si vamos al navegador de archivos veremos los directorios que hemos creado



El primero ejemplo será quitar todos los permisos a un fichero. Utilizaremos la siguiente línea de comando: **chmod a-rwx Test**. Con esto decimos que para todo el mundo *no habrá permisos ni de escritura, ni lectura, ni de ejecución sobre este archivo*. Para esto nos situamos en la carpeta que contiene los documentos.



A continuación veremos que los permisos fueron asignados correctamente.



Como podemos ver en la figura, Daniel no tiene acceso al archivo debido a los permisos configurados anteriormente.



Ahora al mismo archivo vamos a aplicarle permisos de solo lectura para todos los usuarios. La línea de comando a utilizar será la siguiente: **chmod a+r Test**.

```
root@localhost:/home/Daniel/Data
Archivo Editar Ver Terminal Solapas Ayuda
[root@localhost ~]# cd /home
[root@localhost home]# cd Daniel
[root@localhost Daniel]# cd Data
[root@localhost Data]# ls -l
total 16
drwxr-xr-x  2 root root 4096 feb  5 02:18 Prueba
d-----  2 root root 4096 feb  5 02:18 Test
[root@localhost Data]#
[root@localhost Data]# chmod a+r Test
[root@localhost Data]#
[root@localhost Data]# ls -l
total 16
drwxr-xr-x  2 root root 4096 feb  5 02:18 Prueba
dr--r--r--  2 root root 4096 feb  5 02:18 Test
[root@localhost Data]# chmod a+rwx Test/
[root@localhost Data]# ls -l
total 16
drwxr-xr-x  2 root root 4096 feb  5 02:18 Prueba
drwxrwxrwx  2 root root 4096 feb  5 02:18 Test
[root@localhost Data]#
```

Como vemos al definir el permiso **a+r** se le aplico permisos de solo lectura para todos los usuarios.

Si establecemos el valor **a+rwx** todo el mundo tendrá permiso de lectura, escritura y ejecución.

Para finalizar con la aplicación de permisos por medio de caracteres haremos otro ejemplo, en el cual vamos a permitir que el grupo al cual pertenezca el fichero tenga acceso a leer y escribir, y los otros no tendrán acceso a ejecutarlo.

Si cambiamos los permisos a un directorio y deseamos que estos permisos tengan efecto sobre todos sus subdirectorios y archivos sólo deberemos añadir la opción **-R**.  
**Ejemplo: \$ chmod a=rw Moreno -R.**

Ahora aplicaremos permisos utilizando valores numéricos. Cada permiso tiene asignado un valor:

**r = 4** (lectura)

**w = 2** (escritura)

**x = 1** (ejecución)

**- = 0** (sin permisos)



Cuando asignamos los permisos por números, primero se sumarán los valores. **La siguiente tabla contiene la suma de los permisos:**

Valor	Permisos	Descripción
0	---	El valor cero significa que no se han asignado permisos
1	--x	sólo se ha asignado el de ejecución
2	-w-	sólo permiso de escritura
3	-wx	permisos de escritura y ejecución
4	r--	sólo permiso de lectura
5	r-x	permisos de lectura y ejecución
6	rw-	permisos de lectura y escritura
7	rxw	permisos: lectura, escritura y ejecución

Los permisos por números se asignan **en grupos de 3**, es decir, para el **propietario-grupo-otros**, no es factible asignar solo para uno o dos de ellos.

Existen varias combinaciones interesantes, estas son:

**rw-rw-rw- (666)** — Todo el mundo puede leer y escribir en el archivo. ***¡No es una buena elección!***

**rxwxrwx (777)** — Todo el mundo puede leer, escribir y ejecutar. ***¡Tampoco es buena elección!***

**rw----- (600)** — Sólo el propietario tiene el derecho de leer y escribir.

A continuación le asignaremos **permisos de escritura, lectura y ejecución** para el propietario del fichero. Al grupo le **asignaremos permisos de lectura y escritura**, y al otro **solo permiso de lectura**.

```
root@localhost:/home/Daniel/Data
Archivo Editar Ver Terminal Solapas Ayuda
[root@localhost Data]# ls
Prueba Test
[root@localhost Data]# ls -l
total 16
drwxrwxr-- 2 root root 4096 feb  5 02:18 Prueba
drwxrwxrwx 2 root root 4096 feb  5 02:18 Test
[root@localhost Data]# chmod 764 Test/
[root@localhost Data]#
[root@localhost Data]# ls -l
total 16
drwxrwxr-- 2 root root 4096 feb  5 02:18 Prueba
drwxrw-r-- 2 root root 4096 feb  5 02:18 Test
[root@localhost Data]#
```

Ahora en el siguiente ejemplo, quitaremos todos los permisos de un fichero a todos los usuarios incluyendo el propietario. Para esto se utiliza **la suma de valores 000**.

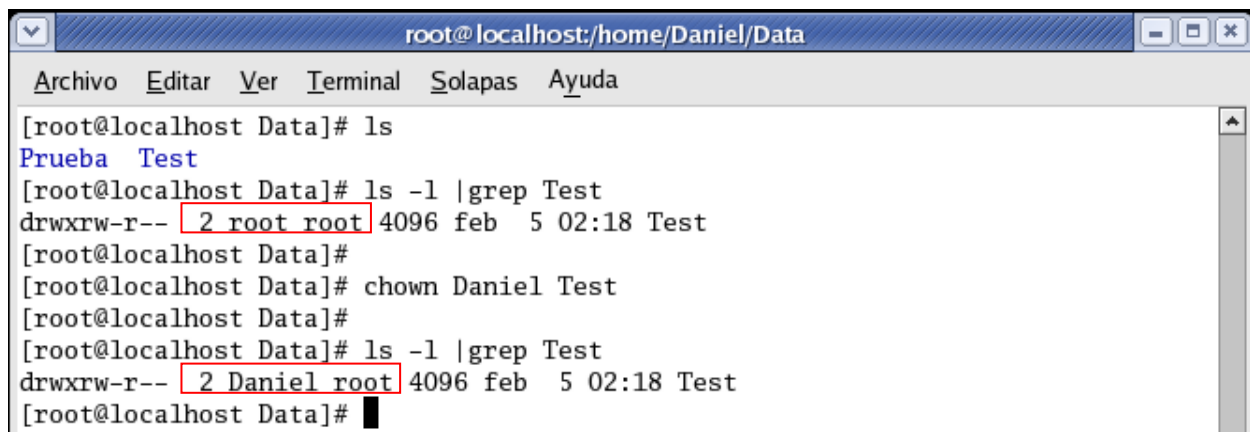
```
root@localhost:/home/Daniel/Data
Archivo Editar Ver Terminal Solapas Ayuda
[root@localhost Data]# chmod 000 Prueba
[root@localhost Data]#
[root@localhost Data]# ls -l
total 16
d----- 2 root root 4096 feb  5 02:18 Prueba
drwxrw-r-- 2 root root 4096 feb  5 02:18 Test
[root@localhost Data]#
```

## Cambiar el propietario

Para cambiar el propietario de un fichero utilizamos el comando **chown**, seguido del **nombre del usuario** al cual pertenecerá el fichero, y por último **el nombre del fichero**.

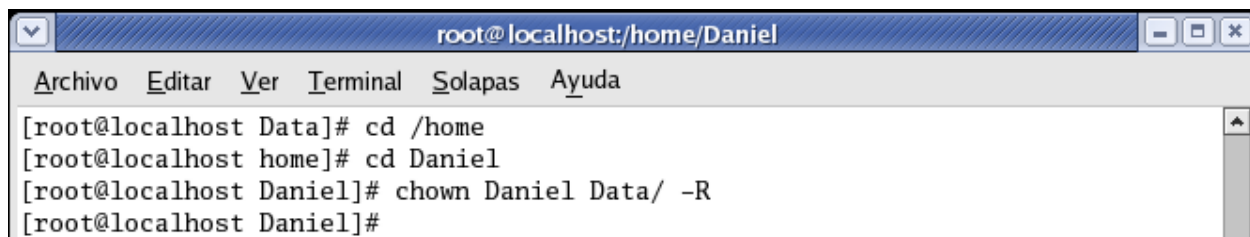
**Chown Ranelson Test** permitirá que el archivo **Test** ahora pertenezca al usuario **Daniel**, aunque haya sido **creado por root**.

Si utilizamos el **comando ls -l | grep Test** veremos que pertenece a root, luego pertenecerá a **Daniel**.



```
root@localhost:/home/Daniel/Data
Archivo Editar Ver Terminal Solapas Ayuda
[root@localhost Data]# ls
Prueba Test
[root@localhost Data]# ls -l |grep Test
drwxrw-r-- 2 root root 4096 feb  5 02:18 Test
[root@localhost Data]#
[root@localhost Data]# chown Daniel Test
[root@localhost Data]#
[root@localhost Data]# ls -l |grep Test
drwxrw-r-- 2 Daniel root 4096 feb  5 02:18 Test
[root@localhost Data]#
```

En el siguiente ejemplo, haremos que el **usuario Ranelson** se convierta en *propietario de todo un directorio*, incluyendo los subdirectorios que incluya. Utilizaremos la siguiente línea de comandos: **chown Daniel documentos/ -R**.



```
root@localhost:/home/Daniel
Archivo Editar Ver Terminal Solapas Ayuda
[root@localhost Data]# cd /home
[root@localhost home]# cd Daniel
[root@localhost Daniel]# chown Daniel Data/ -R
[root@localhost Daniel]#
```

De esta forma hemos terminado de trabajar con **usuarios, creación de grupos y asignación de permisos**.