

Práctica de gestion de usuarios y permisos

Christian Amauri

Benemérita Universidad Autónoma de Puebla
Facultad de ciencias de la computación
Av. San claudio y 14 sur, puebla méxico.
christian.amadoro@alumno.buap.mx

En esta práctica vamos a documentar los pasos necesarios a seguir (junto con sus respectivos comandos de linux) para la gestion de usuarios y los posibles permisos que estos pueden tener sobre diversas áreas o características en una red. En la sección de introducción damos un poco de preámbulo de cómo se desarrolla la práctica y los materiales que necesitamos para realizarla. En la sección de Desarrollo se documentan todos los pasos es los que consiste la práctica. En la sección de conclusiones hacemos una pequeña reflexión sobre el trabajo realizado y en la sección de bibliografía y referencias tenemos las fuentes consultadas y material de apoyo.

1 Introduccion

En el mundo de la tecnología, Linux es uno de los sistemas operativos más utilizados y apreciados. Una de las razones de su popularidad es su capacidad para administrar usuarios y permisos de forma precisa y flexible. En este informe, exploraremos los comandos de gestión de usuarios en Linux, que son herramientas fundamentales para gestionar usuarios y grupos de usuarios en un sistema Linux. Aprenderemos cómo crear y eliminar usuarios, establecer y modificar permisos de usuario y grupo, y cómo verificar la información de los usuarios. Conocer estos comandos es esencial para cualquier administrador de sistemas que busque una gestión de usuarios eficiente y segura.

Objetivo

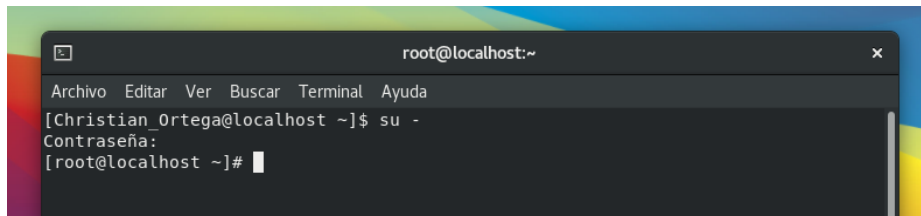
Comprender cómo funcionan los diferentes permisos para los usuarios y qué campos afectan . Además de obvio, conocer los comandos necesarios para llevar a cabo su modificación

Materiales

- Computadora con CentOS instalado
- Terminal de Linux

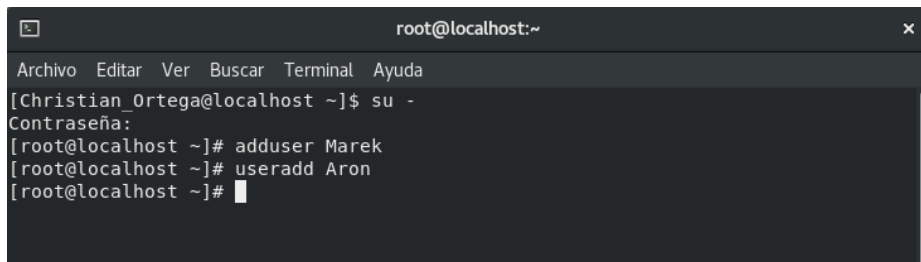
2 Desarrollo

Primero entramos al modo root con el commando su -:



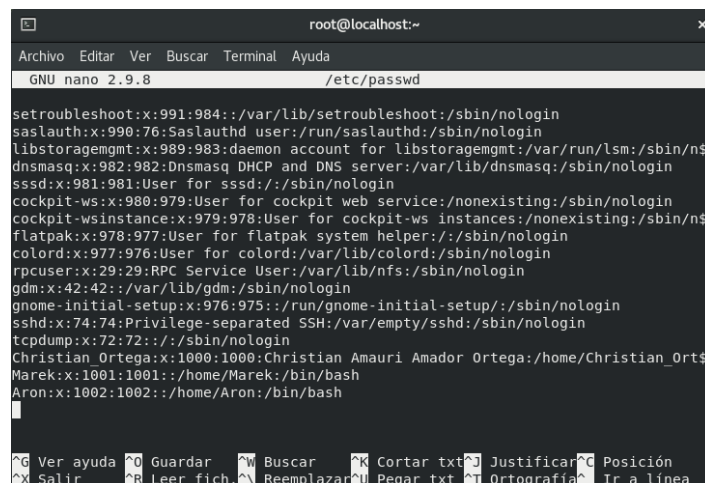
```
root@localhost:~  
Archivo Editar Ver Buscar Terminal Ayuda  
[Christian_Ortega@localhost ~]$ su -  
Contraseña:  
[root@localhost ~]#
```

Ahora añadimos unos cuantos usuarios, esto lo Podemos hacer mediante adduser, o mediante useradd, seguido del nombre que deseamos para el nuevo usuario:



```
root@localhost:~  
Archivo Editar Ver Buscar Terminal Ayuda  
[Christian_Ortega@localhost ~]$ su -  
Contraseña:  
[root@localhost ~]# adduser Marek  
[root@localhost ~]# useradd Aron  
[root@localhost ~]#
```

Verificamos que los nuevos usuarios fueron creados mediante nano /etc/passwd:



```
root@localhost:~  
Archivo Editar Ver Buscar Terminal Ayuda  
GNU nano 2.9.8 /etc/passwd  
setroubleshoot:x:991:984::/var/lib/setroubleshoot:/sbin/nologin  
sasauth:x:990:76:Sasauthd user:/run/sasauthd:/sbin/nologin  
libstoragemgmt:x:989:983:daemon account for libstoragemgmt:/var/run/lsm:/sbin/n$  
dnsmasq:x:982:982:Dnsmasq DHCP and DNS server:/var/lib/dnsmasq:/sbin/nologin  
sssd:x:981:981:User for sssd:/:/sbin/nologin  
cockpit-ws:x:980:979:User for cockpit web service:/nonexisting:/sbin/nologin  
cockpit-wsinstance:x:979:978:User for cockpit-ws instances:/nonexisting:/sbin/n$  
flatpak:x:978:977:User for flatpak system helper:/:/sbin/nologin  
colord:x:977:976:User for colord:/var/lib/colord:/sbin/nologin  
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin  
gdm:x:42:42:/var/lib/gdm:/sbin/nologin  
gnome-initial-setup:x:976:975:/run/gnome-initial-setup:/sbin/nologin  
sshd:x:74:74:Privilege-separated SSH:/var/empty/ssh:/sbin/nologin  
tcpdump:x:72:72:/:/sbin/nologin  
Christian Ortega:x:1000:1000:Christian Amauri Amador Ortega:/home/Christian_Ort$  
Marek:x:1001:1001:/home/Marek:/bin/bash  
Aron:x:1002:1002:/home/Aron:/bin/bash  
^G Ver ayuda ^O Guardar ^W Buscar ^K Cortar txt ^J Justificar ^C Posición  
^X Salir ^R Leer fich. ^U Reemplazar ^V Pegar txt ^T Ortografía ^_ Ir a línea
```

Ahora verificamos el estado de los grupos mediante nano /etc/group:

```
root@localhost:~  
Archivo Editar Ver Buscar Terminal Ayuda  
[Christian_Ortega@localhost ~]$ su -  
Contraseña:  
[root@localhost ~]# adduser Marek  
[root@localhost ~]# useradd Aron  
[root@localhost ~]# nano /etc/passwd  
[root@localhost ~]# nano /etc/passwd  
[root@localhost ~]# nano /etc/group
```

```
GNU nano 2.9.8 /etc/group  
  
flatpak:x:977:  
colord:x:976:  
rpcuser:x:29:  
gdm:x:42:  
gnome-initial-setup:x:975:  
sshd:x:74:  
slocate:x:21:  
tcpdump:x:72:  
Christian_Ortega:x:1000:  
Marek:x:1001:  
Aron:x:1002:  
  
^G Ver ayuda ^O Guardar ^W Buscar ^K Cortar txt ^J Justificar ^C Posición  
^X Salir ^R Leer fich. ^E Reemplazar ^U Pegar txt ^T Ortografía ^I Ir a línea
```

Para cada usuario Podemos verificar una dirección de grupo mediante el uso del comando grep -nombre de usuario- /etc/group:

```
root@localhost:~  
Archivo Editar Ver Buscar Terminal Ayuda  
[Christian_Ortega@localhost ~]$ su -  
Contraseña:  
[root@localhost ~]# adduser Marek  
[root@localhost ~]# useradd Aron  
[root@localhost ~]# nano /etc/passwd  
[root@localhost ~]# nano /etc/passwd  
[root@localhost ~]# nano /etc/group  
[root@localhost ~]# grep Marek /etc/passwd  
Marek:x:1001:1001:./home/Marek:/bin/bash  
[root@localhost ~]# grep Marek /etc/group  
Marek:x:1001:  
[root@localhost ~]#
```

Para crear carpetas completas, tenemos la siguiente sintáxis, donde “armandito” es la carpeta y “Armando” es el usuario:

```
root@localhost:~  
Archivo Editar Ver Buscar Terminal Ayuda  
[root@localhost ~]# useradd -m -d /home/armandito -e 20231006 armando  
[root@localhost ~]#
```

Y volvemos a usar el commando grep, pero con la sintáxis del uso de una carpeta:

```
root@localhost:~  
Archivo Editar Ver Buscar Terminal Ayuda  
[root@localhost ~]# useradd -m -d /home/armandito -e 20231006 armando  
[root@localhost ~]# grep armandito /etc/passwd  
armando:x:1003:1003::/home/armandito:/bin/bash  
[root@localhost ~]#
```

El commando chage nos sirve para definir una fecha de expiación para un usuario

```
root@localhost:~  
Archivo Editar Ver Buscar Terminal Ayuda  
[root@localhost ~]# chage -E 20230515 armando  
[root@localhost ~]#
```

Mediante el commando ls (desde home) verificamos los usuarios que tenemos disponibles.

```
root@localhost:/home  
Archivo Editar Ver Buscar Terminal Ayuda  
[root@localhost ~]# chage -E 20230515 armando  
[root@localhost ~]# cd /home  
[root@localhost home]# ls  
armandito Aron Christian_Ortega Marek  
[root@localhost home]#
```

Mediante el commando -l Podemos obtener información sobre las fechas de expiración de una cuenta:

```
root@localhost:/
Archivo Editar Ver Buscar Terminal Ayuda
[root@localhost ~]# chage -E 20230515 armando
[root@localhost ~]# cd /home
[root@localhost home]# ls
armandito Aron Christian Ortega Marek
[root@localhost home]# cd /
[root@localhost /]# chage -l armando
Último cambio de contraseña : feb 23, 2023
La contraseña caduca : nunca
Contraseña inactiva : nunca
La cuenta caduca : abr 09, 57359
Número de días mínimo entre cambio de contraseña : 0
Número de días máximo entre cambio de contraseña : 99999
Número de días de aviso antes de que caduque la contraseña : 7
[root@localhost /]#
```

Ahora, para añadir grupos, usamos el commando groupadd, seguido del nombre del grupo que queremos definir:

```
root@localhost:/
Archivo Editar Ver Buscar Terminal Ayuda
[root@localhost /]# groupadd ADMREDES
[root@localhost /]#
```

Y así Podemos modificar el grupo de un usuario:

```
[root@localhost /]# usermod -G ADMREDES Aron
[root@localhost /]#

[root@localhost /]# passwd Aron
Cambiando la contraseña del usuario Aron.
Nueva contraseña:
Vuelva a escribir la nueva contraseña:
passwd: todos los tokens de autenticación se actualizaron exitosamente.
[root@localhost /]#
```

Si ejecutamos el comando `pwck /etc/passwd`. Obtenemos el siguiente resultado:

```
root@localhost:~  
Archivo Editar Ver Buscar Terminal Ayuda  
[root@localhost ~]# pwck /etc/passwd  
usuario «pipewire»: el directorio «/var/run/pipewire» no existe  
usuario «pulse»: el directorio «/var/run/pulse» no existe  
usuario «clevis»: el directorio «/var/cache/clevis» no existe  
usuario «gluster»: el directorio «/run/gluster» no existe  
usuario «saslauth»: el directorio «/run/saslauthd» no existe  
usuario «cockpit-ws»: el directorio «/nonexisting» no existe  
usuario «cockpit-wsinstance»: el directorio «/nonexisting» no existe  
usuario «gnome-initial-setup»: el directorio «/run/gnome-initial-setup/» no existe  
pwck: sin cambios  
[root@localhost ~]#
```

Ahora hacemos lo mismo con el comando `pwck /etc/shadow` :

```
root@localhost:~  
Archivo Editar Ver Buscar Terminal Ayuda  
[root@localhost ~]# pwck /etc/passwd  
usuario «pipewire»: el directorio «/var/run/pipewire» no existe  
usuario «pulse»: el directorio «/var/run/pulse» no existe  
usuario «clevis»: el directorio «/var/cache/clevis» no existe  
usuario «gluster»: el directorio «/run/gluster» no existe  
usuario «saslauth»: el directorio «/run/saslauthd» no existe  
usuario «cockpit-ws»: el directorio «/nonexisting» no existe  
usuario «cockpit-wsinstance»: el directorio «/nonexisting» no existe  
usuario «gnome-initial-setup»: el directorio «/run/gnome-initial-setup/» no existe  
pwck: sin cambios  
[root@localhost ~]# pwck /etc/shadow  
entrada del fichero de contraseñas incorrecta  
¿desea eliminar la línea «root:$6$BgipKUIgyPjTbP0G$u0RKgI2AMhUbPLMyCsIIW.NG0wY/u  
6XrhUx26dnmMMNZcT4cBXXgYiAa8zYWNodJwpdvMuNoyLRQawCghDzYk0::0:99999:7:::»? █
```

En este caso, nos muestra un mensaje de “entrada del fichero de contraseñas incorrecta”. Seguido de muchas preguntas de “desea eliminar la línea <...?”. Si esto ocurre, intentemos escribir el comando `pwconv`. E intentar nuevamente el proceso anterior, nos debería salir algo parecido a la pantalla de ejecución del comando `pwck /etc/passwd`.

Ahora, creamos un usuario llamado LUCAS, en el grupo NUEVOSUSUARIOS. En `home/LUCAS`, con programa de inicio: `/bin/sh` mediante las siguientes líneas de comando:

```
[root@localhost ~]# groupadd NUEVOSUSUARIOS  
[root@localhost ~]# sudo useradd -m -d /home/LUCAS -s /bin/sh -G NUEVOSUSUARIOS LUCAS  
[root@localhost ~]# █
```

Comprobamos los cambios mediante el comando su - <nombre del usuario (en este caso, LUCAS)> :

```
LUCAS
[root@localhost ~]# su - LUCAS
[LUCAS@localhost ~]$
```

Ahora, cambiamos algo de su información de usuario:

```
[root@localhost Christian_Ortega]# sudo usermod -c "LUCAS ADDAMS, FCC, 2224961605" -d /home/LUCAS-o LUCAS
[root@localhost Christian_Ortega]#
```

Ahora creamos un grupo llamado CLASERED, y añadimos a LUCAS a él:

```
Christian_Ortega@localhost:/home/Christian_Ortega
Archivo Editar Ver Buscar Terminal Ayuda
[root@localhost Christian_Ortega]# groupadd CLASERED
[root@localhost Christian_Ortega]# gpasswd -a LUCAS CLASERED
Añadiendo al usuario LUCAS al grupo CLASERED
[root@localhost Christian_Ortega]#
```

Creamos un usuario libre y lo añadimos también:

```
[root@localhost Christian_Ortega]# useradd JotchuaNojao
[root@localhost Christian_Ortega]# gpasswd -a JotchuaNojao CLASERED
Añadiendo al usuario JotchuaNojao al grupo CLASERED
[root@localhost Christian_Ortega]#
```

Ahora, mediante el comando `id -u` e `id -g` vamos a verificar los UID y GID de los usuarios “root”, “PACO” y “LUCAS”:

```
Christian_Ortega@localhost:/home/Christian_Ortega
Archivo Editar Ver Buscar Terminal Ayuda
[root@localhost Christian_Ortega]# id -u root
0
[root@localhost Christian_Ortega]# id -g root
0
[root@localhost Christian_Ortega]# id -u PACO
1008
[root@localhost Christian_Ortega]# id -g PACO
1011
[root@localhost Christian_Ortega]# id -u LUCAS
1006
[root@localhost Christian_Ortega]# id -g LUCAS
1008
[root@localhost Christian_Ortega]#
```

Ahora, bloquearemos la cuenta de lucas mediante el comando `-l`:

```
Christian_Ortega@localhost:/home/Christian_Ortega
Archivo Editar Ver Buscar Terminal Ayuda
[root@localhost Christian_Ortega]# passwd -l LUCAS
Bloqueando la contraseña para el usuario LUCAS.
passwd: Éxito
[root@localhost Christian_Ortega]# su LUCAS
sh-4.4$ nano /etc/passwd
sh-4.4$
```

(podemos notar que ahora el prompt es distinto)

Ahora volvemos a activar la cuenta de LUCAS, hay que tener en cuenta que para este proceso la cuenta que estamos manejando debe tener definida una contraseña, porque puede haber inconvenientes en el camino de caso contrario. En la siguiente captura ,podemos ver que nos aseguramos de que LUCAS tenga una contraseña para que el comando `-U` pueda funcionar apropiadamente.

```
[root@localhost Christian_Ortega]# passwd LUCAS
Cambiando la contraseña del usuario LUCAS.
Nueva contraseña:
Vuelva a escribir la nueva contraseña:
passwd: todos los tokens de autenticación se actualizaron exitosamente.
[root@localhost Christian_Ortega]# usermod -U LUCAS
[root@localhost Christian_Ortega]#
```


PARTE II: Modificar permisos:

Creamos 9 archivos mediante el comando touch:

```
root@localhost:~  
Archivo Editar Ver Buscar Terminal Ayuda  
[root@localhost ~]# touch arch3  
[root@localhost ~]# touch arch4  
[root@localhost ~]# touch arch5  
[root@localhost ~]# touch arch6  
[root@localhost ~]# touch arch7  
[root@localhost ~]# touch arch8  
[root@localhost ~]# touch arch9  
[root@localhost ~]#
```

Verificamos los permisos que tienen, mediante el comando ls -l:

```
root@localhost:~  
Archivo Editar Ver Buscar Terminal Ayuda  
[root@localhost ~]# touch arch3  
[root@localhost ~]# touch arch4  
[root@localhost ~]# touch arch5  
[root@localhost ~]# touch arch6  
[root@localhost ~]# touch arch7  
[root@localhost ~]# touch arch8  
[root@localhost ~]# touch arch9  
[root@localhost ~]# ls -l  
total 8  
-rw-r--r--. 1 root root    0 mar 13 01:26 ach1  
-rw-----. 1 root root 1009 ene 28 13:42 anaconda-ks.cfg  
-rw-r--r--. 1 root root    0 mar 13 01:26 arch2  
-rw-r--r--. 1 root root    0 mar 13 01:27 arch3  
-rw-r--r--. 1 root root    0 mar 13 01:27 arch4  
-rw-r--r--. 1 root root    0 mar 13 01:27 arch5  
-rw-r--r--. 1 root root    0 mar 13 01:27 arch6  
-rw-r--r--. 1 root root    0 mar 13 01:27 arch7  
-rw-r--r--. 1 root root    0 mar 13 01:27 arch8  
-rw-r--r--. 1 root root    0 mar 13 01:27 arch9  
-rw-r--r--. 1 root root 1436 ene 28 20:15 initial-setup-ks.cfg  
[root@localhost ~]#
```

Modificamos los permisos de los archivos mediante el comando chmod:

```
root@localhost:~  
Archivo Editar Ver Buscar Terminal Ayuda  
-rw-r--r--. 1 root root 0 mar 13 01:26 ach1  
-rw-----. 1 root root 1009 ene 28 13:42 anaconda-ks.cfg  
-rw-r--r--. 1 root root 0 mar 13 01:26 arch2  
-rw-r--r--. 1 root root 0 mar 13 01:27 arch3  
-rw-r--r--. 1 root root 0 mar 13 01:27 arch4  
-rw-r--r--. 1 root root 0 mar 13 01:27 arch5  
-rw-r--r--. 1 root root 0 mar 13 01:27 arch6  
-rw-r--r--. 1 root root 0 mar 13 01:27 arch7  
-rw-r--r--. 1 root root 0 mar 13 01:27 arch8  
-rw-r--r--. 1 root root 0 mar 13 01:27 arch9  
-rw-r--r--. 1 root root 1436 ene 28 20:15 initial-setup-ks.cfg  
[root@localhost ~]# chmod u=rwx ach1  
[root@localhost ~]# chmod u=rw arch2  
[root@localhost ~]# chmod a=rwx arch3  
[root@localhost ~]# chmod u=rwx,g=rw,o=r arch4  
chmod: modo inválido: «u=rwx,g=rw,o=r»  
Pruebe 'chmod --help' para más información.  
[root@localhost ~]# chmod u=rwx,g=rw,o=r arch4  
[root@localhost ~]# chmod u=rwx,g=r arch5  
[root@localhost ~]# chmod u=r,x,g=rw,o=r arch6  
[root@localhost ~]# chmod u=r,o=x arch7  
[root@localhost ~]# chmod u=rw,g=r,o=x arch8  
[root@localhost ~]# chmod u=rw,g=rw,o=x arch9  
[root@localhost ~]#
```

```
root@localhost:~  
Archivo Editar Ver Buscar Terminal Ayuda  
[root@localhost ~]# ls -l  
total 8  
-rwxr--r--. 1 root root 0 mar 13 01:26 ach1  
-rw-----. 1 root root 1009 ene 28 13:42 anaconda-ks.cfg  
-rw-r--r--. 1 root root 0 mar 13 01:26 arch2  
-rwxrwxrwx. 1 root root 0 mar 13 01:27 arch3  
-rwxrw-r--. 1 root root 0 mar 13 01:27 arch4  
-rwxr--r--. 1 root root 0 mar 13 01:27 arch5  
-r-xrw-r--. 1 root root 0 mar 13 01:27 arch6  
-r--r---x. 1 root root 0 mar 13 01:27 arch7  
-rw-r---x. 1 root root 0 mar 13 01:27 arch8  
-rw-rw---x. 1 root root 0 mar 13 01:27 arch9  
-rw-r--r--. 1 root root 1436 ene 28 20:15 initial-setup-ks.cfg  
[root@localhost ~]#
```

Ahora, modificamos los permisos anteriores, con la metodología de la sintaxis del operador + y -:

```
root@localhost:~  
Archivo Editar Ver Buscar Terminal Ayuda  
-rw-rw--x. 1 root root    0 mar 13 01:27 arch9  
-rw-r--r--. 1 root root 1436 ene 28 20:15 initial-setup-ks.cfg  
[root@localhost ~]# chmod o+r ach1  
[root@localhost ~]# chmod u-w arch2  
[root@localhost ~]# chmod a-x arch3  
[root@localhost ~]# chmod g-r,o-r arch4  
[root@localhost ~]# chmod g-r,o+wr arch5  
[root@localhost ~]# chmod u+w,o-r arch6  
[root@localhost ~]# chmod u+w,g+x,o+w-x arch8  
[root@localhost ~]# chmod u+x,g-r-w,o+r arch9
```

```
root@localhost:~  
Archivo Editar Ver Buscar Terminal Ayuda  
-rw-rw--x. 1 root root    0 mar 13 01:27 arch9  
-rw-r--r--. 1 root root 1436 ene 28 20:15 initial-setup-ks.cfg  
[root@localhost ~]# chmod o+r ach1  
[root@localhost ~]# chmod u-w arch2  
[root@localhost ~]# chmod a-x arch3  
[root@localhost ~]# chmod g-r,o-r arch4  
[root@localhost ~]# chmod g-r,o+wr arch5  
[root@localhost ~]# chmod u+w,o-r arch6  
[root@localhost ~]# chmod u+w,g+x,o+w-x arch8  
[root@localhost ~]# chmod u+x,g-r-w,o+r arch9  
[root@localhost ~]# ls -l  
total 8  
-rwxr--r--. 1 root root    0 mar 13 01:26 ach1  
-rw-----. 1 root root 1009 ene 28 13:42 anaconda-ks.cfg  
-r--r--r--. 1 root root    0 mar 13 01:26 arch2  
-rw-rw-rw-. 1 root root    0 mar 13 01:27 arch3  
-rwx-w----. 1 root root    0 mar 13 01:27 arch4  
-rwx---rw-. 1 root root    0 mar 13 01:27 arch5  
-rwxrw----. 1 root root    0 mar 13 01:27 arch6  
-r--r----x. 1 root root    0 mar 13 01:27 arch7  
-rw-r-x-w-. 1 root root    0 mar 13 01:27 arch8  
-rwx---r-x. 1 root root    0 mar 13 01:27 arch9  
-rw-r--r--. 1 root root 1436 ene 28 20:15 initial-setup-ks.cfg  
[root@localhost ~]#
```

Ahora creamos otros nueve archivos más, mediante el comando touch, y procedemos a cambiar sus permisos mediante la sintáxis octal:

```
root@localhost:~  
Archivo Editar Ver Buscar Terminal Ayuda  
[root@localhost ~]# touch b1  
[root@localhost ~]# touch b2  
[root@localhost ~]# touch b3  
[root@localhost ~]# touch b4  
[root@localhost ~]# touch b5  
[root@localhost ~]# touch b6  
[root@localhost ~]# touch b7  
[root@localhost ~]# touch b8  
[root@localhost ~]# touch b9  
[root@localhost ~]#
```

```
root@localhost:~  
Archivo Editar Ver Buscar Terminal Ayuda  
[root@localhost ~]# touch b8  
[root@localhost ~]# touch b9  
[root@localhost ~]# chmod 421 b1  
[root@localhost ~]# chmod 000 b2  
[root@localhost ~]# chmod 777 b3  
[root@localhost ~]# chmod 564 b4  
[root@localhost ~]# chmod 740 b5  
[root@localhost ~]# chmod 644 b6  
[root@localhost ~]# chmod 645 b7  
[root@localhost ~]# chmod 764 b8  
[root@localhost ~]# chmod 700 b9
```

Verificamos los cambios:

```
root@localhost:~  
Archivo Editar Ver Buscar Terminal Ayuda  
[root@localhost ~]# chmod 700 b9  
[root@localhost ~]# ls -l  
total 8  
-rwxr--r--. 1 root root 0 mar 13 01:26 ach1  
-rw----- 1 root root 1009 ene 28 13:42 anaconda-ks.cfg  
-r--r--r--. 1 root root 0 mar 13 01:26 arch2  
-rw-rw-rw-. 1 root root 0 mar 13 01:27 arch3  
-rwx-w---. 1 root root 0 mar 13 01:27 arch4  
-rwx---rw-. 1 root root 0 mar 13 01:27 arch5  
-rwxrw---- 1 root root 0 mar 13 01:27 arch6  
-r--r---X. 1 root root 0 mar 13 01:27 arch7  
-rw-r-x-w-. 1 root root 0 mar 13 01:27 arch8  
-rwx---r-X. 1 root root 0 mar 13 01:27 arch9  
-r--w---X. 1 root root 0 mar 13 01:51 b1  
----- 1 root root 0 mar 13 01:52 b2  
-rwxrwxrwx. 1 root root 0 mar 13 01:52 b3  
-r-xrw-r--. 1 root root 0 mar 13 01:52 b4  
-rwxr----- 1 root root 0 mar 13 01:52 b5  
-rw-r--r--. 1 root root 0 mar 13 01:52 b6  
-rw-r--r-X. 1 root root 0 mar 13 01:52 b7  
-rwxrw-r--. 1 root root 0 mar 13 01:52 b8  
-rwx----- 1 root root 0 mar 13 01:52 b9  
-rw-r--r--. 1 root root 1436 ene 28 20:15 initial-setup-ks.cfg  
[root@localhost ~]#
```

Ahora, con una sola instrucción, quitamos los permisos de lectura, escritura y ejecución para “otros” a todos los archivos usados en el ultimo ejercicio, mediante el operador “*” (en seguida mostramos los cambios). Así:

```
root@localhost:~  
Archivo Editar Ver Buscar Terminal Ayuda  
[root@localhost ~]# chmod o-rwx b*  
[root@localhost ~]# ls -l  
total 8  
-rwxr--r--. 1 root root    0 mar 13 01:26 ach1  
-rw-----. 1 root root 1009 ene 28 13:42 anaconda-ks.cfg  
-r--r--r--. 1 root root    0 mar 13 01:26 arch2  
-rw-rw-rw-. 1 root root    0 mar 13 01:27 arch3  
-rwx-w----. 1 root root    0 mar 13 01:27 arch4  
-rwx---rw-. 1 root root    0 mar 13 01:27 arch5  
-rwxrw----. 1 root root    0 mar 13 01:27 arch6  
-r--r----x. 1 root root    0 mar 13 01:27 arch7  
-rw-r-x-w-. 1 root root    0 mar 13 01:27 arch8  
-rwx---r-x. 1 root root    0 mar 13 01:27 arch9  
-r---w----. 1 root root    0 mar 13 01:51 b1  
-----. 1 root root    0 mar 13 01:52 b2  
-rwxrwx---. 1 root root    0 mar 13 01:52 b3  
-r-xrw----. 1 root root    0 mar 13 01:52 b4  
-rwxr-----. 1 root root    0 mar 13 01:52 b5  
-rw-r-----. 1 root root    0 mar 13 01:52 b6  
-rw-r-----. 1 root root    0 mar 13 01:52 b7  
-rwxrw----. 1 root root    0 mar 13 01:52 b8  
-rwx-----. 1 root root    0 mar 13 01:52 b9  
-rw-r--r--. 1 root root 1436 ene 28 20:15 initial-setup-ks.cfg  
[root@localhost ~]#
```

3 Conclusiones

En conclusión, los comandos de gestión de usuarios en Linux son herramientas poderosas y fundamentales para administrar usuarios y grupos de usuarios en un sistema Linux. A través de este informe, hemos aprendido cómo crear y eliminar usuarios, establecer y modificar permisos de usuario y grupo, y verificar información de usuario. Es importante destacar que la gestión de usuarios adecuada y segura es esencial para garantizar la integridad y seguridad del sistema. Por lo tanto, conocer y dominar estos comandos es una habilidad esencial para cualquier administrador de sistemas que busque una gestión de usuarios eficiente y segura en Linux.

4 Bibliografía y referencias

- <https://keepcoding.io/blog/gestion-de-usuarios-y-grupos-en-linux/>
- https://www.linuxtotal.com.mx/index.php?cont=info_admon_008
- Clines, M., & Dean, M. (2019). Linux Administration: The Linux Operating System and Command Line Guide for Linux Administrators. Packt Publishing Ltd.
- Ranjan, R., & Shah, R. (2018). Hands-On Linux Administration on Azure: Explore virtual machines, scale sets, and more to manage Linux workloads on Azure. Packt Publishing Ltd.
- Bhatia, R. (2015). Network Administrators Survival Guide: Networking Protocols, Security, and Virtualization. Apress.