Linux para administradores (intermedio)

Manuel Domínguez

Bienvenidos!

En esta sección corresponde con la **Gestión de usuarios**.

Y en esta clase, vamos a explicar cómo se gestionan **las cuentas de usuario y grupos.**

1.- Introducción

Una de las **funciones principales** de un administrador, es sin duda, gestionar las cuentas de usuarios. Vamos a ver:

Mostrar información de las cuentas de usuarios y grupos.
Mostrar información sobre los usuarios que están conectados en el sistema.
Crear cuentas de usuarios y grupos.
Modificar cuentas de usuarios.
Activar/desactivar cuentas de usuarios.
Eliminar cuenta de usuarios y grupos.

2.- Mostrar información de las cuentas de usuario.

El usuario se caracteriza por:

```
usuario@debian:~$ id
uid=1000(usuario) gid=1000(usuario) grupos=1000(usuario),24(cdrom),25(floppy),29(audio),30(
dip),44(video),46(plugdev),109(netdev),112(bluetooth),116(lpadmin),117(scanner)
usuario@debian:~$ id -u
1000
usuario@debian:~$ id root
uid=0(root) gid=0(root) grupos=0(root)
usuario@debian:~$ id -u root
0
```

Username: Nombre de usuario

UID: Identificador de usuario.

GID´s: Grupos a los que pertenece

2.- Mostrar información de las cuentas de usuario.

La información de las cuentas de usuario se guarda en dos ficheros:

/etc/passwd
Usuarios del sistema

/etc/shadow
Las contraseñas de los usuarios y su caducidad.

```
usuario:x:1000:1000:usuario,,,:/home/usuario:/bin/bash
systemd-coredump:x:999:999:systemd Core Dumper:/:/usr/
vboxadd:x:998:1::/var/run/vboxadd:/bin/false
sshd:x:117:65534::/run/sshd:/usr/sbin/nologin
heidi:x:1001:1001:,,,:/home/heidi:/bin/bash
pedro:x:1002:1002:,,,:/home/pedro:/bin/bash
```

```
usuario:$6$jCf/QgKqBtrgLJNw$9MNLpe9hwU4msz
MuMNVsDEZ7gRimofpSwr401:18286:0:99999:7:::
systemd-coredump:!!:18286::::::
vboxadd:!:18300::::::
sshd:*:18364:0:99999:7:::
heidi:$6$uSGWhuOOHH6gndGi$NluASXUyHv4U4kht
L9Qkq55nFJ9wsCwQe/OQ0:18366:0:99999:7:::
pedro:$6$H9Ki6NiI8.5kWfkl$6C/w5hXWeWiWi9qX
XlLj/X5a.Qv6jU6jSDeB.:18366:0:99999:7:::
```

2.- Mostrar información de las cuentas de usuario.

/etc/passwd → Contiene la lista de usuarios definidos por el sistema. (man 5 passwd)

Nombre:password:uid:gid:gecos:home:shell

nombre: nombre de usuario
password: contraseña. Si aparece una x significa que la contraseña está encriptada y se guarda en /etc/shadow
uid: identificador de usuario
gid: identificador del grupo principal.
gecos: información secundaria del usuario:Nombre, tlf, etc
home: Directorio de trabajo.
Shell: Intérprete de comandos.

2.- Mostrar información de las cuentas de usuario.

Práctica: /etc/passwd → Obtener información de un usuario

1 .- Obtenemos información del usuario usuario

 # cat /etc/passwd|grep "^usuario"

2.- Obtenemos información del usuario, usuario.

#getent passwd usuario → Es mucho más fácil y rápido

2.- Mostrar información de las cuentas de usuario.

Práctica: /etc/passwd → Significa de la x

1.- Quitemos al usuario (usuario), la x en el campo password, para ver qué ocurre.

2.- Cerramos la sesión y nos volvemos a loguear.

3.- Para dejarlo como estaba, volvemos a colocar la x en su lugar.

2.- Mostrar información de las cuentas de usuario.

/etc/shadow → Contiene las contraseñas encriptadas de los usuarios, así como información sobre la caducidad de la cuenta y contraseña (man shadow).

nombre:password:changed:min:max:warn:inactive:expired

nombre: nombre de usuario
password: contraseña encriptada: MD5, SHA256, SHA512, etc
changed: La última vez que se ha cambiado la contraseña. Se expresa en días, contados desde 1 de enero de 1970.
min: Nº de días que han de pasar para poder cambiar la contraseña.
max: Nºde días que puede permanecer con la misma contraseña.
warn: Nº de días que nos avisa antes de que caduque la contraseña.
inactive: Nº de días que va a esperar, una vez que ha caducado la cuenta, antes de deshabilitar la cuenta.
expired: Fecha en la que la cuenta expira.

2.- Mostrar información de las cuentas de usuario.

Práctica:/etc/shadow → Obtener información de cuenta del usuario

1.- Obtener información de la caducidad de la cuenta y contraseña del usuario (usuario), a partir del fichero /etc/shadow.

También podrás utilizar: **#getent shadow usuario**

2.- Ejecuta el comando: **#chage -l usuario**, e interpreta el resultado, comparándola con la información anterior.

2.- Mostrar información de las cuentas de usuario.

Usuarios conectados:

w → IDLE: El tiempo que lleva de inactividad. FROM: Equipo remoto

who → Equivalente al w, pero nos da menos información.

users → Listados de usuarios conectados.

lastlog → Información de la última vez que se ha logeado los usuarios.

lastb → Usuarios que han fallado a loguearse.

3.- Crear cuentas de usuarios.

Para crear cuentas de usuarios podemos utilizar:

adduser → es una forma rápida, fácil e **interactiva** de crear usuarios.

useradd → es otra forma, pero **no es interactiva**. **Ideal para crear cuentas de usuarios de forma masiva.**

En esta clase, nos centraremos en el adduser y en la clase de creación/eliminación masiva de usuarios, nos centraremos en useradd.

3.- Crear cuentas de usuarios.

Ficheros de configuración
Existen unos ficheros donde se guardan los valores por defecto en la creación de las cuentas:

/etc/default/useradd → Podemos configurar **EXPIRE, INACTIVE**

/etc/login.defs → #cat /etc/login.defs |grep -v "^#\|^$"

```
PASS_MAX_DAYS   99999
PASS_MIN_DAYS   0
PASS_WARN_AGE   7

#
# Min/max values for automatic
#
UID_MIN                  1000
UID_MAX                 60000
# System accounts
#SYS_UID_MIN              100
#SYS_UID_MAX              999
```

INACTIVE=-1
Significa que la cuenta no será bloqueada cuando expire la contraseña.

3.- Crear cuentas de usuarios.

Ficheros de inicialización:

/etc/skel → Se guardan unos ficheros que se copian al directorio HOME del usuario, cuando se crea una cuenta.

Por ejemplo:

.bashrc → Cada vez que se ejecuta una shell.

.bash_logout → Al salir del sistema

.profile → Al hacer un login

3.- Crear cuentas de usuarios.

Práctica:/etc/skel → Pasos previos

1.- Creamos dos cuentas nuevas: heidi (heidi) y pedro (pedro)

2.- Veámos que se han introducido en /etc/passwd y /etc/shadow

3.- Comprobamos la información con:
getent passwd heidi #getent shadow heidi

4.- Creamos dentro de /mnt un directorio, llamado sugerencias y le damos permisos para que todos los usuarios puedan escribir y sólo borrar el dueño del archivo. **#chmod 1777 /mnt/sugerencias.**

3.- Crear cuentas de usuarios.

Práctica: /etc/skel → Escribimos dentro

5.- **Dentro de /etc/skel:**

.- Creamos dentro de /etc/skel, un archivo llamado **LEEME**:
"Bienvenidos al sistema. Respeta las normas establecidas"

.- Creamos un enlace blando a la carpeta /mnt/sugerencias:
ln -s /mnt/sugerencias /etc/skel

6.- Creamos un usuario nuevo: **abuelo** y vemos lo que hay en su interior.

4.- Modificar cuentas de usuarios.

Para modificar las cuentas de usuarios, podemos utilizar varios comandos:

passwd → Principalmente es utilizado para cambiar la contraseña, aunque también se puede utilizar para modificar las características de la cuenta.

usermod → modificar cuentas de usuario.

chage → modificar cuentas de usuario.

4.- Modificar cuentas de usuarios.

Práctica: passwd

1.- Le cambiamos al usuario abuelo la contraseña.
#passwd abuelo

2.- Vemos el estado de la cuenta (opción -S):

```
root@servidor:~# passwd -S abuelo
abuelo P 04/17/2020 0 99999 7 -1
```

#passwd -S abuelo
Primer campo: nombre de usuario
2º y 3º campos: [**P: Activa L:Bloqueada**] Fecha del último cambio de contraseña
Los 4 siguientes:: min,max,warn,inactive (días)

4.- Modificar cuentas de usuarios.

Práctica: usermod→ Bloquear y desbloquear una cuenta

1.- Vamos a bloquear la cuenta del usuario abuelo.

#usermod -L abuelo

2.- Si intentamos loguearnos, veremos que no podemos.

3.- Vamos a ver lo que ha ocurrido:
#getent shadow abuelo → !antes de la contraseña
#passwd -S abuelo → Nos dice cuando fue bloqueada.

4.- Desbloqueemos el usuario: **#usermod -U abuelo**

4.- Modificar cuentas de usuarios.

Práctica: usermod → Cambiar el nombre de usuario

1.- Vamos a cambiar el nombre de usuario al usuario abuelo.

Comprobamos que no esté conectado: #who
Comprobamos sus datos:#getent passwd abuelo

#usermod -l Nombre_Nuevo Nombre_Actual
#usermod -l abuelito abuelo

2.- Comprobamos que sólo ha cambiado el nombre:
#getent passwd abuelo

3.- Lo dejamos como estaba.

4.- Modificar cuentas de usuarios.

chage → # chage [opciones] usuario

```
Opciones:
  -d, --lastday ULTIMO_DÍA      establece el último cambio de clave a
                                ULTIMO_DÍA
  -E, --expiredate FECHA_EXP    establece la fecha de caducidad de la
                                cuenta a FECHA_EXP
  -h, --help                    muestra este mensaje de ayuda y termina
  -I, --inactive INACTIV        desactiva la cuenta después de INACTIV
                                días desde la fecha de expiración
  -l, --list                    muestra la información de envejecimiento
                                de la cuenta
  -m, --mindays DÍAS_MIN        establece el número mínimo de días antes
                                de cambiar la clave a DÍAS_MIN
  -M, --maxdays DÍAS_MAX        establece el número máximo de días antes
                                de cambiar la clave a DÍAS_MAX
  -W, --warndays DÍAS_AVISO     establece el número de días de aviso
                                a DÍAS_AVISO
```

4.- Modificar cuentas de usuarios.

Práctica: chage

Vamos hacer unos cambios sobre la cuenta: heidi

1.-**#chage -d 0 heidi** → va a obligar a cambiar la contraseña cuando se loguee.

2.- Que la cuenta del usuario expire el 30 de junio. → YYYY-MM-DD

#chage -E 2020-06-30 heidi

3.- Comprobación: **#getent shadow hedi**

contínua

4.- Modificar cuentas de usuarios.

Práctica: chage

4.- Viajamos al 1 de julio y intentamos entrar como heidi.

#systemctl networking stop → Para que no se actualice la fecha
#date --set='2020-07-01'

5.- Intentamos entrar como heidi.

6.- Desactivar la expiración de la cuenta: **#chage E -1 usuario**

7.- Volvemos activar la red y si es necesario solicitamos una ip: **#dhclient.**

5.- Eliminar cuentas de usuario

userdel: Elimina la cuenta de usuario.

#userdel [opciones] nombre_usuario

Opción muy interesante:

-r → Elimina también el directorio Home del usuario.

Ejemplo:

1.- Borramos la cuenta del usuario abuelo, y su directorio Home.

6.- Información de grupos.

La información de los grupos se guarda en dos ficheros:

/etc/group
Grupos del sistema

```
usuario:x:1000:
systemd-coredump:x:999:
heidi:x:1001:
pedro:x:1002:
abuelo:x:1003:
profesor:x:1004:
```

/etc/gshadow
Las contraseñas de los grupos

```
usuario:!::
systemd-coredump:!!::
heidi:!::
pedro:!::
abuelo:!::
profesor:!::
```

6.- Información de grupos.

/etc/group

Grupo:x:GID:usuarios

Grupo: Nombre del grupo

x: La existencia de un fichero gshadow, donde se guardan las contraseñas.

GID: grupo principal

Usuarios: usuarios que tienen ese grupo como secundario.

6.- Información de grupos.

/etc/gshadow :

Grupo:Contraseña:Administrador:Usuarios.

No se suele poner contraseñas a los grupos, pero es importante para proteger, al grupo.

Grupo: Nombre del grupo.
Clave: Contraseña encriptada.
! → Ningún usuario puede añadirse al grupo por su cuenta (newgroup)
* → Está deshabilitado.
Administrador: Es el administrador del grupo
Usuarios: usuarios que pertenecen al grupo.

6.- Información de grupos.

Práctica: Grupos de un usuario

1.- Podemos ver los grupos a los que pertenece un usuario:

#groups usuario
#groups heidi

7.- Gestión de grupos

Para la gestión de grupos podemos utilizar:

#addgroup grupo **#groupadd grupo** → Crear un nuevo grupo

#groupmod grupo → Modificar un grupo existente

#groupdel grupo → eliminar un grupo

#adduser usuario grupo → Añadir un usuario al grupo.

#deluser usuario grupo → Quitar un usuario de un grupo.

#newgrp grupo → Especifica cuál es el grupo principal de un usuario.

7.- Gestión de grupos

Práctica: grupos

1.- Creamos **un grupo llamado analista: #groupadd analista**

2.- Comprobamos:

#getent group analista

#getent gshadow analista

3.- Introducimos el usuario heidi y pedro al grupo analista.

#adduser heidi analista **#adduser pedro analista**

4.- Comprobamos: #getent group analista #getent gshadow analista

7.- Gestión de grupos

Práctica: grupos

Cambiar el grupo principal de heidi.

1.- Heidi crea un archivo proyecto1 → grupo heidi

2.- Heidi establece su grupo principal:

heidi$ **newgrp analista** → Le estamos diciendo que su grupo principal sea analista.

3.- Heidi crea un archivo proyecto2 → grupo analista

8.- Eliminar un usuario de un grupo y un grupo.

Práctica: grupos

1.- Vemos los grupos a los que pertenece heidi. #groups heidi

2.- Lo quitamos del grupo analista: #deluser heidi analista

3.- Eliminamos el grupo analista: #groupdel analista

9.- Desconectar a un usuario

Práctica: Desconectar a un usuario

.- Si necesitamos desconectar a un usuario, podemos utilizar:

#pkill -9 -u heidi

Realmente, lo que hemos hecho es matar todos los procesos de un usuario.

RETO

Práctica:

1.- Crea un script: **UsuariosBloqueados.sh**, que nos muestre los usuarios del sistema que tienen la cuenta bloqueada.

Pista: Utiliza el resultado del comando: #passwd -S usuario
Si sale L significa que está bloqueada.

RETO

Práctica:

2.- Queremos que los usuarios que lleven más de 30 minutos =1800 s sin actividad, se les cierre la sesión.

Pista: La variable TMOUT define el tiempo que queremos permitir a un usuario permanecer dentro de la shell o sesión SSH inactivo.

Para probarlo:
1.- Como root: Puedes poner TMOUT=60
2.- Loguearte como heidi en un terminal y dejar 60 s para ver qué ocurre.

Linux para administradores (intermedio)

Manuel Domínguez

Despedida

Hemos llegado al final de este vídeo..

Nos vemos en el siguiente.