

Instructivo Desafío 4

1) Crear 3 usuarios (Asegurarse de que sus directorios home sean creados)

a) Mariano b) Luis c) Jose

Abrimos una terminal

Ejecutamos los siguientes comandos (-m crea el home del usuario)

a) 'sudo useradd -m Mariano'

b) 'sudo useradd -m Luis'

c) 'sudo useradd -m Jose'

Adjunto captura:

```
● desafio4@desafio4:~$ sudo useradd -m Mariano
● desafio4@desafio4:~$ sudo useradd -m Luis
● desafio4@desafio4:~$ sudo useradd -m Jose
```

Compruebo que estén creados los directorios home

Adjunto captura:

```
● desafio4@desafio4:~$ cd /home
● desafio4@desafio4:/home$ ls
desafio4  Jose  Luis  Mariano
```

2) Crear 3 grupos a) Administradores b) Desarrolladores c) Operaciones

Ejecutamos los siguientes comandos en la terminal

a) 'sudo groupadd Administradores'

b) 'sudo groupadd Desarrolladores'

c) 'sudo groupadd Operaciones'

Adjunto captura:

```
● desafio4@desafio4:~$ sudo groupadd Administradores
[sudo] password for desafio4:
● desafio4@desafio4:~$ sudo groupadd Desarrolladores
● desafio4@desafio4:~$ sudo groupadd Operaciones
```

3) Realizar las siguientes modificaciones:

- a) Mariano tiene que ser parte del grupo Administradores (primario)
- b) Luis tiene que ser parte del grupo Desarrolladores (primario)
- c) Jose tiene que ser parte del grupo Administradores (primario) y Operaciones (Secundario)

Ejecutamos los siguientes comandos en la terminal

- a) `'sudo usermod -g Administradores Mariano'`
- b) `'sudo usermod -g Desarrolladores Luis'`
- c) `'sudo usermod -g Administradores -G Operaciones Jose'`

(aclaración `'-g'` grupo primario `'-G'` grupo secundario)

Adjunto captura:

```
● desafio4@desafio4:~$ sudo usermod -g Administradores Mariano
[sudo] password for desafio4:
● desafio4@desafio4:~$ sudo usermod -g Desarrolladores Luis
● desafio4@desafio4:~$ sudo usermod -g Administradores -G Operaciones Jose
```

Adjunto comprobación utilizando `'groups'` (Nos dice en qué grupos está un usuario)

```
● desafio4@desafio4:~$ groups Luis
Luis : Desarrolladores
● desafio4@desafio4:~$ groups Mariano
Mariano : Administradores
● desafio4@desafio4:~$ groups Jose
Jose : Administradores Operaciones
```

4) Los miembros del grupo de Administradores, deben poder correr cualquier comando como sudo sin utilizar la contraseña

Ejecutamos los siguientes comandos en la terminal

`'sudo visudo'` (luego ingresamos la password, por defecto se abre con nano)

Luego buscamos la línea `'%sudo ALL=(ALL:ALL) ALL'`

Debajo agregamos esta línea `'%Administradores ALL=(ALL:ALL) NOPASSWD:ALL'`

Adjunto captura:

```
# Allow members of group sudo to execute any command
%sudo  ALL=(ALL:ALL) ALL
%Administradores  ALL=(ALL:ALL) NOPASSWD:ALL
# See sudoers(5) for more information on "@include" directives:
```

Luego presionamos **'ctrl + O'** y nos pregunta si deseamos guardar, apretamos **'enter'**
Y salimos apretando **'ctrl + X'**

5) Los miembros del grupo Operaciones, deben poder correr el comando mount y umount (con cualquier argumento) como sudo y tendrán que utilizar la contraseña.

Ejecutamos los siguientes comandos en la terminal

'sudo visudo' (luego ingresamos la password, por defecto se abre con nano)

Luego al final del archivo agregamos la siguiente línea:

'%Operaciones ALL=(ALL) PASSWD: /bin/mount, /bin/umount'

Adjunto captura:

```
# See sudoers(5) for more information on "@include" directives:
%Operaciones ALL=(ALL) PASSWD: /bin/mount, /bin/umount
@includedir /etc/sudoers.d
```

Luego presionamos **'ctrl + O'** y nos pregunta si deseamos guardar, apretamos **'enter'**
Y salimos apretando **'ctrl + X'**

6) Desarrollar la siguiente estructura de directorios y archivos:

a) Dentro del Home del usuario Mariano, tendrán que crear la siguiente estructura:

i) Scripts/ping.sh (tendrá que tener el siguiente contenido) (permisos 2770, owner Mariano y grupo Administradores) `#!/bin/bash ping www.google.com.ar`

Primero vamos a cambiar la contraseña para poder logear con el usuario Mariano:

Ejecutamos en la terminal los siguientes comandos:

Ejecutamos **'sudo passwd Mariano'** (e ingresamos la nueva password y la repetimos)

Luego cambiamos de usuario usando **'su Mariano'** (e ingresamos la password)

Y comprobamos utilizando el **'whoami'**

Adjunto captura:

```
desafio4@desafio4:~$ su Mariano
Password:
$ whoami
Mariano
$
```

Primero creamos los directorios (Scripts/docs)

Ejecutamos `'cd /home/Mariano'`

Luego `'mkdir Scripts'`

`'mkdir docs'`

Captura de la creación y `'ls'` para mostrar los directorios nuevos

```
$ cd /home/Mariano
$ mkdir Scripts
$ mkdir docs
$ ls
Desktop      Downloads    Public      Templates
docs         Music       Scripts     Videos
Documents    Pictures    snap
```

i) Luego `'nano Scripts/ping.sh'`

Y editamos el archivo

```
GNU nano 6.2
#!/bin/bash
ping www.google.com.ar
```

Luego presionamos `'ctrl + O'` y nos pregunta si deseamos guardar, apretamos enter

Y salimos apretando `'ctrl + X'`

Ahora cambiamos los permisos

Usamos `'chmod 2770 Scripts/ping.sh'`

`'chown Mariano:Administradores Scripts/ping.sh'`

Adjunto captura:

```
$ chmod 2770 Scripts/ping.sh
$ chown Mariano:Administradores Scripts/ping.sh
```

ii) Scripts/kick.sh (permiso rwx-rwx-, owner Mariano y grupo administradores)

Creamos el archivo ejecutando: **'touch Scripts/kick.sh'**

Adjunto captura **'ls'** con los dos archivos creados (ping.sh y kick.sh)

```
$ cd Scripts
$ ls
kick.sh  ping.sh
$
```

ii) Ejecutamos los siguientes comandos

Ejecutamos **'chmod u=rwx,g=rwx,o= Scripts/kick.sh'**

'chown Mariano:Administradores Scripts/kick.sh'

```
$ chmod u=rwx,g=rwx,o= Scripts/kick.sh
$ chown Mariano:Administradores Scripts/kick.sh
```

iii) docs/test.txt (permiso el owner puede leer y escribir, el grupo puede leer y escribir y los demás usuarios no pueden hacer nada, owner Mariano y grupo administradores)

iv) docs/readme.txt (permiso 660, owner Mariano y grupo Administradores)

En esta caso cree los 2 archivos al mismo tiempo

Ejecutamos **'touch docs/test.txt'**

'touch docs/readme.txt'

Adjunto captura de comandos, y **'ls'** para listar

```
$ touch docs/test.txt
$ touch docs/readme.txt
$ cd docs
$ ls
readme.txt  test.txt
```

iii) Luego ejecutamos los siguientes comandos:

Ejecutamos `'chmod u=rw,g=rw,o= docs/test.txt'`
`'chown Mariano:Administradores docs/test.txt'`

Adjunto captura del comando:

```
$ chmod u=rw,g=rw,o= docs/test.txt
$ chown Mariano:Administradores docs/test.txt
```

iv) Luego ejecutamos los siguientes comandos:

Ejecutamos `'chmod 660 docs/readme.txt'`
`'chown Mariano:Administradores docs/readme.txt'`

Adjunto captura:

```
$ chmod 660 docs/readme.txt
$ chown Mariano:Administradores docs/readme.txt
```

b) Dentro del Home del usuario Luis, tendrán que crear la siguiente estructura:

i) feat-1/index.html (tendrá que tener el siguiente contenido) (permisos ugo=664, owner Luis y grupo Luis)
<H1> Hola Bootcamp! </H1>

Primero vamos a cambiar la contraseña para poder logear con el usuario Luis:

Ejecutamos en la terminal los siguientes comandos:

Ejecutamos `'sudo passwd Luis'` (e ingresamos la nueva password y la repetimos)

Luego cambiamos de usuario usando `'su Luis'` (e ingresamos la password)

Y comprobamos utilizando el `'whoami'`

Adjunto captura:

```
desafio4@desafio4:/$ su Luis
Password:
$ whoami
Luis
```

Primero creamos los directorio (feat-1)

Ejecutamos `'cd /home/Luis'`

Luego `'mkdir feat-1'`

Adjunto captura del directorio creado:

```
$ cd /home/Luis
$ mkdir feat-1
$ ls
feat-1
```

i) Luego ejecutamos los siguientes comandos

Creamos el archivo usando el comando

`'echo "<H1> Hola bootcamp! </H1>" > /home/Luis/feat-1/index.html'`

Entramos al directorio haciendo `'cd feat-1'`

Hacemos un `'ls'` para listar y ver el nuevo archivo creado

Por último un `'cat index.html'`, para ver el contenido del archivo

Adjunto captura de salida de comandos:

```
$ echo "<H1> Hola Bootcamp! </H1>" > /home/Luis/feat-1/index.html
$ ls
Desktop    feat-1    Public    Videos
Documents  Music     snap
Downloads  Pictures  Templates
$ cd feat-1
$ ls
index.html
$ cat index.html
<H1> Hola Bootcamp! </H1>
```

Ahora vamos a configurar los permisos.

Ejecutamos los siguientes comandos:

`'chmod 664 feat-1/index.html'`

`'chown Luis:Luis feat-1/index.html'`

Adjunto captura de salida de comandos:

```
$ chmod 664 feat-1/index.html
$ chown Luis:Luis feat-1/index.html
chown: changing ownership of 'feat-1/index.html': Operation not permitted
$ su desafio4
Password:
desafio4@desafio4:/home/Luis$ sudo chown Luis:Luis feat-1/index.html
```

Para cambiar el propietario y el grupo de un archivo o directorio sin otorgar privilegios de sudo a 'Luis', utilice mi otro usuario llamado 'desafio4'

Adjunto captura de comando 'ls -l'

```
$ ls -l
total 4
-rw-rw-r-- 1 Luis Luis 26 jul 20 16:54 index.html
$
```

ii) feat-1/readme.txt (permisos rwrwr, owner Luis y grupo Desarrolladores)

Primero vamos a crear el archivo readme.txt

Para eso ejecutamos este comando para ir al directorio '`cd /home/Luis/feat-1`'
Luego creamos el archivo haciendo un '`touch readme.txt`'

Por último cambiamos lo permisos

Ejecutamos los siguientes comandos:

`'chmod u-rw,g=rw,o=r feat-1/readme.txt'`

`chown Luis:Desarrolladores feat-1/readme.txt'`

Adjunto captura de salida de comandos:

```
$ chmod u=rw,g=rw,o=r feat-1/readme.txt
$ chown Luis:Desarrolladores feat-1/readme.txt
```

Adjunto captura de comando 'ls -l'

```
$ cd feat-1
$ ls -l
total 4
-rw-rw-r-- 1 Luis Luis 26 jul 20 16:54 index.html
-rw-rw-r-- 1 Luis Desarrolladores 0 jul 20 17:03 readme.txt
```


iii) feat-2/tests (permisos rwrwr, owner Luis y grupo Luis)

Primero verificamos haciendo un **'ls'** que estamos en directorio el home de luis
Luego creamos el directorio y el subdirectorio usando el comando

'mkdir -p feat-2/tests'

Luego hacemos un **'ls'** y verificamos que se creó correctamente el directorio **'feat-2'**
Hacemos un **'cd'** (y verificamos se creó el directorio tests)

Adjunto captura de salida de comandos:

```
$ ls
Desktop    feat-1    Public    Videos
Documents  Music     snap
Downloads  Pictures  Templates
$ mkdir -p feat-2/tests
$ ls
Desktop    feat-1    Pictures  Templates
Documents  feat-2    Public    Videos
Downloads  Music     snap
$ cd feat-2
$ ls
tests
$
```

Por último cambiamos los permisos:

Ejecutamos **'chmod u=rw,g=rw,o=r feat-2/tests'**

'chown Luis:Luis feat-2/tests' (este nos tira error)

```
$ chmod u=rw,g=rw,o=r feat-2/tests
$ chown Luis:Luis feat-2/tests
chown: changing ownership of 'feat-2/tests': Operation not permitted
$ su desafio4
Password:
desafio4@desafio4:/home/Luis$ sudo chown Luis:Luis feat-2/tests
```

Para cambiar el propietario y el grupo de un archivo o directorio sin otorgar privilegios de sudo a 'Luis', utilice mi otro usuario llamado 'desafio4'

Adjunto captura del comando **'ls -l'**

```
$ ls -l
total 4
drw-rw-r-- 2 Luis Luis 4096 jul 20 17:10 tests
```

- c) Dentro del Home del usuario Jose, tendrán que crear la siguiente estructura:
i) ops/script.sh (permisos 4770, owner Jose y grupo operaciones)

Primero vamos a cambiar la contraseña para poder logear con el usuario Jose:
Ejecutamos en la terminal los siguientes comandos:
Ejecutamos **'sudo passwd Jose'** (e ingresamos la nueva password y la repetimos)
Luego cambiamos de usuario usando **'su Jose'** (e ingresamos la password)
Y comprobamos utilizando el **'whoami'**

Adjunto captura:

```
desafio4@desafio4:~$ su Jose
Password:
$ whoami
```

Primero vamos al directorio /home/Jose usando el comando **'cd /home/Jose'**
Luego creamos el directorio usando **'mkdir ops'**
Luego ingresamos al directorio ops usando **'cd ops'**
Y creamos el archivo usando **'touch script.sh'**
Comprobamos con **'ls'** que el archivo está creado.

Adjunto captura:

```
$ cd /home/Jose
$ ls
Desktop    Music      snap
Documents  Pictures   Templates
Downloads  Public     Videos
$ mkdir ops
$ ls
Desktop    Music      Public     Videos
Documents  ops         snap
Downloads  Pictures    Templates
$ cd ops
$ touch script.sh
$ ls
script.sh
```

Ahora vamos a editar los permisos

Ejecutamos: `'chmod 4770 ops/script.sh'`

`chown Jose:Operaciones ops/script.sh'`

```
$ chmod 4770 ops/script.sh
$ chown Jose:Operaciones ops/script.sh
```

Adjunto captura de `'ls -l'`

```
$ ls -l
total 0
-rwsrwx--- 1 Jose Operaciones 0 jul 20 17:27 script.sh
```

ii) ops/test.sh (permisos el owner tiene todos los permisos, el grupo también y los demás usuarios sólo pueden leer, owner Jose y grupo operaciones)

Dentro del directorio ops

Ejecutamos el comando `'touch test.sh'` para crear el archivo

Ahora editamos los permisos

Ejecutamos `'chmod u=rwx,g=rwx,o=r ops/test.sh'`

`'chown Jose:Operaciones ops/test.sh'`

```
$ chmod u=rwx,g=rwx,o=r ops/test.sh
$ chown Jose:Operaciones ops/test.sh
```

Adjunto captura del comando `'ls -l'` (comprobando permisos)

```
$ ls -l
total 0
-rwsrwx--- 1 Jose Operaciones 0 jul 20 17:27 script.sh
-rwxrwxr-- 1 Jose Operaciones 0 jul 20 17:43 test.sh
```

iii) adm/mount.sh (Tendrá que tener el siguiente contenido) (permisos rwxrwx, owner Jose y grupo administradores, además agregar un ACL que haga que el usuario Mariano no pueda ejecutar este archivo, sólo pueda leerlo).

```
#!/bin/bash
```

```
mount -a
```

Primero vamos al directorio /home/Jose usando el comando `'cd /home/Jose'`

Luego creamos el directorio usando `'mkdir adm'`

Luego ingresamos al directorio adm usando `'cd adm'`

Y creamos el archivo usando `'nano mount.sh'`

Luego agregamos el contenido al archivo `#!/bin/bash`
`mount -a`

Verificamos con un `'ls'` que el archivo está creado

Adjunto captura:

```
$ cd /home/Jose
$ mkdir adm
$ cd adm
$ nano mount.sh
$ ls
mount.sh
```

Luego presionamos `'ctrl + O'` y nos pregunta si deseamos guardar, apretamos enter
Y salimos apretando `'ctrl + X'`

Ahora vamos a modificar los permisos

Ejecutamos `'chmod u=rwx,g=rwx,o= adm/mount.sh'`

`'chown Jose:Administradores adm/mount.sh'`

```
$ chmod u=rwx,g=rwx,o= adm/mount.sh
$ chown Jose:Administradores adm/mount.sh
```

Adjunto captura del comando `'ls -l'` (comprobando permisos)

```
$ ls -l
total 4
-rwxrwx--- 1 Jose Administradores 21 jul 20 17
:59 mount.sh
```

Por último vamos a agregar un ACL (que haga que el usuario Mariano no pueda ejecutar este archivo, sólo pueda leerlo)

Primero ejecutamos el comando **'getfacl mount.sh'** (para ver como esta configurado el ACL)

Adjunto captura de salida de comando:

```
$ getfacl mount.sh
# file: mount.sh
# owner: Jose
# group: Administradores
user::rwx
group::rwx
other::---
```

Ahora vamos a modificarlo

Ejecutamos **'setfacl -m u:Mariano:r mount.sh'**

Luego hacemos un **'getfacl mount.sh'** (para verificar configuración)

Adjunto captura:

```
$ setfacl -m u:Mariano:r mount.sh
$ getfacl mount.sh
# file: mount.sh
# owner: Jose
# group: Administradores
user::rwx
user:Mariano:r--
group::rwx
mask::rwx
other::---
```

iv) procedures/kick.txt (permisos rwrwr, owner Jose y grupo operaciones)

Primero vamos al directorio /home/Jose usando el comando **'cd /home/Jose'**

Luego creamos el directorio usando **'mkdir procedures'**

Luego ingresamos al directorio adm usando **'cd procedures'**

Por último hacemos un **'touch kick.txt'**

Y comprobamos con un **'ls'**

Adjunto captura:

```
$ cd /home/Jose
$ mkdir procedures
$ ls
adm      Documents  Music  Pictures  Public  Templates
Desktop  Downloads  ops    procedures snap    Videos
$ cd procedures
$ touch kick.txt
$ ls
kick.txt
```

Ahora vamos a configurar los permisos:

Ejecutamos **'chmod u=rw,g=rw,o=r procedures/kick.txt'**
'chown Jose:Operaciones procedures/kick.txt'

Adjunto captura:

```
chmod u=rw,g=rw,o=r procedures/kick.txt
chown Jose:Operaciones procedures/kick.txt
```

Adjunto captura del comando **'ls -l'** (comprobando permisos)

```
$ cd procedures
$ ls -l
total 0
-rw-rw-r-- 1 Jose Operaciones 0 jul 21 21:35 kick.txt
```

7) Bloquear el acceso del usuario Luis (no borrar el usuario, solo bloquear el acceso)

Ejecutamos el comando **'sudo passwd -l Luis'** (Con esta opción **'-l'**, el administrador del sistema puede inhabilitar la contraseña de algún usuario específico.)

Adjunto captura:

```
desafio4@desafio4:~$ sudo passwd -l Luis
[sudo] password for desafio4:
passwd: password expiry information changed.
desafio4@desafio4:~$
```

Ahora probamos que no podemos loguear con el usuario Luis
Usamos el comando '**su Luis**' (e ingresamos la contraseña)
Y nos da error

Adjunto captura:

```
desafio4@desafio4:~$ su Luis
Password:
su: Authentication failure
```

8) Forzar un cambio de contraseña en el siguiente login del usuario Jose

Ejecutamos el comando '**sudo passwd -e Jose**' (con '**-e**' Esta opción forzará al usuario a cambiar su contraseña en su siguiente login al sistema.)

Adjunto captura:

```
desafio4@desafio4:~$ sudo passwd -e Jose
passwd: password expiry information changed.
```

Archivo editado

```
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
*/5 * * * * /home/Mariano/Scripts/ping.sh
```

Ahora comprobamos que nos pide cambiar contraseña:

```
desafio4@desafio4:~$ su Jose
Password:
You are required to change your password immediately (administrator enforced).
Changing password for Jose.
Current password:
New password:
Retype new password:
$
$ whoami
Jose
```

(como se aprecia en la imagen, nos pide cambiar la contraseña y luego compruebo con un '**whoami**' ,que estoy logueado con el usuario Jose)

9) Configurar el vencimiento de la contraseña del usuario Mariano (en 7 días)

Primero ejecutamos **'sudo chage -l Mariano'** (para obtener la información del usuario)

Adjunto captura:at

```
desafio4@desafio4:~$ sudo chage -l Mariano
Last password change           : jul 20, 2023
Password expires                : never
Password inactive              : never
Account expires                : never
Minimum number of days between password change : 0
Maximum number of days between password change : 99999
Number of days of warning before password expires : 7
```

Ahora configuramos el vencimiento en 7 días

Ejecutamos **'sudo chage -M 7 Mariano'**

Volvemos a utilizar el **'sudo chage -l Mariano'** (para obtener la información del usuario)

Adjunto captura:

```
desafio4@desafio4:~$ sudo chage -l Mariano
Last password change           : jul 20, 2023
Password expires                : jul 27, 2023
Password inactive              : never
Account expires                : never
Minimum number of days between password change : 0
Maximum number of days between password change : 7
Number of days of warning before password expires : 7
```

Como se observa (ahora dice que la password expira el 27 de julio (7 días))

1) En el crontab del usuario Mariano, agregar una línea que ejecute el script ping.sh cada 5 minutos de cada hora de cada día

Ejecutamos los siguientes comandos

'sudo crontab -e -u Mariano'

Seleccionamos 1 (nano) y agregamos este comando al final del archivo

'*/5 * * * * /home/Mariano/Scripts/ping.sh'

Adjunto captura:

```
desafio4@desafio4:~$ sudo crontab -e -u Mariano
[sudo] password for desafio4:
crontab: installing new crontab
```

Verificamos con el comando `'grep "ping.sh" /var/log/syslog'` el funcionamiento del script

```
desafio4@desafio4:~$ grep "ping.sh" /var/log/syslog
Jul 21 03:45:01 desafio4 CRON[5138]: (Mariano) CMD (/home/Mariano/Scripts/ping.sh)
Jul 21 03:55:02 desafio4 CRON[1308]: (Mariano) CMD (/home/Mariano/Scripts/ping.sh)
Jul 21 04:00:01 desafio4 CRON[10453]: (Mariano) CMD (/home/Mariano/Scripts/ping.sh)
```

2) En el crontab del usuario Jose, agregar una línea que ejecute el comando `echo $USER > procedures/kick.txt` (agregar el path completo) en la hora 18 de cada día de cada mes

Ejecutamos los siguientes comandos

`'sudo crontab -e -u Jose'`

y agregamos este comando al final del archivo

`'0 18 * * * echo $USER > /home/Jose/procedures/kick.txt'`

Adjunto captura:

```
desafio4@desafio4:~$ sudo crontab -e -u Jose
no crontab for Jose - using an empty one
crontab: installing new crontab
```

Archivo editado:

```
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow  command
0 18 * * * echo $USER > /home/Jose/procedures/kick.txt
```

Luego presionamos `'ctrl + O'` y nos pregunta si deseamos guardar, apretamos `'enter'` Y salimos apretando `'ctrl + X'`

Verificamos con el comando `'grep "kick.txt" /var/log/syslog'` el funcionamiento del script

```
desafio4@desafio4:~$ grep "kick.txt" /var/log/syslog
Jul 21 18:00:01 desafio4 CRON[1000147]: (Jose) CMD (echo $USER > /home/Jose/procedures/kick.txt)
```

3) En el crontab general del sistema, agregar una línea que ejecute el comando free -m como el usuario Mariano en el minuto 15, 30, 45 de las horas 6, 12 y 18 el día Lunes de cada mes

4) En el crontab general del sistema, agregar una línea que ejecute el comando df -h como el usuario Jose cada minuto entre las 15:30 y las 15:35 el día 17 del mes.

Ejecutamos los siguientes comandos

```
'sudo crontab -e'
```

y agregamos al final del archivo

```
'15,30,45 6,12,18 * * 1 su -c "free -m" -s /bin/sh Mariano'
```

```
'30-35 15 17 * * su -c "df -h" -s /bin/sh Jose'
```

Adjunto captura:

```
desafio4@desafio4:~$ sudo crontab -e
no crontab for root - using an empty one
crontab: installing new crontab
```

Archivo editado:

```
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow  command
15,30,45 6,12,18 * * 1 su -c "free -m" -s /bin/sh Mariano
30-35 15 17 * * su -c "df -h" -s /bin/sh Jose
```

Luego presionamos **'ctrl + O'** y nos pregunta si deseamos guardar, apretamos **'enter'** Y salimos apretando **'ctrl + X'**

Verificamos con el comando **'grep "free -m" /var/log/syslog'** el funcionamiento del script (edite el comando para que sea día viernes, así compruebo el funcionamiento, luego lo vuelvo a dejar como lo pide en la consigna)

```
desafio4@desafio4:~$ grep "free -m" /var/log/syslog
Jul 21 06:15:01 desafio4 CRON[248091]: (root) CMD (su -c "free -m" -s /bin/sh Mariano)
Jul 21 06:30:01 desafio4 CRON[274633]: (root) CMD (su -c "free -m" -s /bin/sh Mariano)
Jul 21 06:45:01 desafio4 CRON[301241]: (root) CMD (su -c "free -m" -s /bin/sh Mariano)
Jul 21 12:15:01 desafio4 CRON[876575]: (root) CMD (su -c "free -m" -s /bin/sh Mariano)
Jul 21 12:30:01 desafio4 CRON[903198]: (root) CMD (su -c "free -m" -s /bin/sh Mariano)
Jul 21 12:45:01 desafio4 CRON[929778]: (root) CMD (su -c "free -m" -s /bin/sh Mariano)
Jul 21 18:15:01 desafio4 CRON[1000344]: (root) CMD (su -c "free -m" -s /bin/sh Mariano)
Jul 21 18:30:01 desafio4 CRON[1000822]: (root) CMD (su -c "free -m" -s /bin/sh Mariano)
Jul 21 18:45:01 desafio4 CRON[1001309]: (root) CMD (su -c "free -m" -s /bin/sh Mariano)
```

Verificamos con el comando **'grep "df -h" /var/log/syslog'** el funcionamiento del script (edite el comando para que sea día 21 del mes, así compruebo el funcionamiento, luego lo vuelvo a dejar como lo pide en la consigna)

```
desafio4@desafio4:~$ grep "df -h" /var/log/syslog
Jul 21 15:30:01 desafio4 CRON[997306]: (root) CMD (su -c "df -h" -s /bin/sh Jose)
Jul 21 15:31:01 desafio4 CRON[997741]: (root) CMD (su -c "df -h" -s /bin/sh Jose)
Jul 21 15:32:01 desafio4 CRON[998176]: (root) CMD (su -c "df -h" -s /bin/sh Jose)
Jul 21 15:33:01 desafio4 CRON[998605]: (root) CMD (su -c "df -h" -s /bin/sh Jose)
Jul 21 15:34:01 desafio4 CRON[999035]: (root) CMD (su -c "df -h" -s /bin/sh Jose)
Jul 21 15:35:01 desafio4 CRON[999470]: (root) CMD (su -c "df -h" -s /bin/sh Jose)
```

1) Crear el archivo /tmp/usuarios-eliminados.txt

Primero vamos a desbloquear a usuario Luis, para poder agregarlo al usuarios-eliminados.txt

Ejecutamos **'sudo passwd -u Luis'**

Adjunto captura:

```
desafio4@desafio4:~$ sudo sudo passwd -u Luis
passwd: password expiry information changed.
desafio4@desafio4:~$
```

Luego creamos el archivo

Cambiamos al usuario Mariano, haciendo **'su Mariano'**

Ejecutamos **'sudo touch /tmp/usuarios-eliminados.txt'**

'sudo chmod 666 /tmp/usuarios-eliminados.txt' (esta configuración nos permite habilitar la escritura para cualquier usuario)

```
$ su Mariano
Password:
Warning: your password will expire in 5 days.
$ sudo touch /tmp/usuarios-eliminados.txt
$ sudo chmod 666 /tmp/usuarios-eliminados.txt
$ whoami
Mariano
```

a) Logueado como cada usuario, realizar el comando `echo $USER >> /tmp/usuarios-eliminados.txt`

Ejecutamos `'echo $USER >> /tmp/usuarios-eliminados.txt'`
Luego cambiamos de usuario usando `'su'` (y repetimos el paso)

Adjunto captura:

```
$ whoami
Mariano
$ echo $USER >> /tmp/usuarios-eliminados.txt
$ su Jose
Password:
$ whoami
Jose
$ echo $USER >> /tmp/usuarios-eliminados.txt
$ su Luis
Password:
$ whoami
Luis
$ echo $USER >> /tmp/usuarios-eliminados.txt
```

Revisamos que el archivo este escrito:
Ejecutando un `'cat /tmp/usuarios-eliminados.txt'`

Adjunto captura:

```
$ cat /tmp/usuarios-eliminados.txt
Mariano
Jose
Luis
```

- 2) Automatizar la eliminación de los usuarios con el comando AT:
- a) Eliminar el usuario Jose y su home en 5 m
 - b) Eliminar el usuario Luis y su home en 10m
 - c) Eliminar el usuario Mariano y su home en 15m

2) Primero instalamos at

Ejecutamos `'sudo apt-get update'`
`'sudo apt-get install at'`

Hacemos un `'cat /etc/passwd'`

```
Mariano:x:1001:1004::/home/Mariano:/bin/sh
Luis:x:1002:1005::/home/Luis:/bin/sh
Jose:x:1003:1004::/home/Jose:/bin/sh
```

(observamos que todavía está el usuario Jose)

También un `'cd /home'`

```
desafio4@desafio4:~$ cd /home
desafio4@desafio4:/home$ ls
desafio4 Jose Luis Mariano
```

(observamos que todavía está el home del usuario Jose)

- a) Eliminar el usuario Jose y su home en 5 min

Primero nos hacemos root ejecutando `'sudo su'` (ingresando pw root)

Luego Ejecutamos `'at now + 5 min'`

Luego ponemos `'sudo userdel -r Jose'` y apretamos **Ctrl + D**

Adjunto captura:

```
desafio4@desafio4:/root$ sudo su
[sudo] password for desafio4:
root@desafio4:~# at now + 5 min
warning: commands will be executed using /bin/sh
at Sat Jul 22 13:42:00 2023
at> sudo userdel -r Jose<EOT>
job 1 at Sat Jul 22 13:42:00 2023
root@desafio4:~#
```

Luego de 5 minutos hacemos un 'ls'

```
root@desafio4:~# cd /home
root@desafio4:/home# ls
desafio4 Luis Mariano
```

(vemos que no esta la carpeta de Jose)

Luego de 5 minutos hacemos un 'cat /etc/passwd'

```
desafio4:x:1000:1000:desafio4,,,:/home/desafio4:/bin/bash
sshd:x:129:65534::/run/sshd:/usr/sbin/nologin
Mariano:x:1001:1004::/home/Mariano:/bin/sh
Luis:x:1002:1005::/home/Luis:/bin/sh
```

(vemos que usuario Jose no figura en la lista)

Luego hacemos un 'ls'

```
root@desafio4:~# cd /home
root@desafio4:/home# ls
desafio4 Luis Mariano
```

(vemos que tampoco está la carpeta home de Jose)

b) Eliminar el usuario Luis y su home en 10 min

Ejecutamos 'at now + 10 min'

Luego ponemos 'sudo userdel -r Luis' y apretamos **Ctrl + D**

Adjunto captura:

```
root@desafio4:~# at now + 10 min
warning: commands will be executed using /bin/sh
at Sat Jul 22 13:56:00 2023
at> sudo userdel -r Luis<EOT>
job 2 at Sat Jul 22 13:56:00 2023
```

Luego de los 10 minutos hacemos un 'cat /etc/passwd'

```
desafio4:x:1000:1000:desafio4,,,:/home/desafio4:/bin/bash
sshd:x:129:65534::/run/sshd:/usr/sbin/nologin
Mariano:x:1001:1004::/home/Mariano:/bin/sh
```

(vemos que usuario Luis no figura en la lista)

Luego hacemos un 'ls'

```
root@desafio4:/home# ls
desafio4  Mariano
```

(vemos que tampoco está la carpeta home de Luis)

c) Eliminar el usuario Mariano y su home en 15m

Ejecutamos 'at now + 15 min'

Luego ponemos 'sudo userdel -r Mariano' y apretamos Ctrl + D

Adjunto captura:

```
root@desafio4:~# at now + 15 min
warning: commands will be executed using /bin/sh
at Sat Jul 22 14:12:00 2023
at> sudo userdel -r Mariano<EOT>
job 3 at Sat Jul 22 14:12:00 2023
```

Luego de los 15 minutos hacemos un 'cat /etc/passwd'

```
desafio4:x:1000:1000:desafio4,,,:/home/desafio4:/bin/bash
sshd:x:129:65534::/run/sshd:/usr/sbin/nologin
```

Luego hacemos un 'ls'

```
root@desafio4:/home# ls
desafio4
```