

## SEGURIDAD EN SISTEMAS OPERATIVOS

### 4º Grado en Informática – Complementos de Ing. del Software Curso 2018-19

#### Práctica [1]

#### Sesión [1]

Autor<sup>1</sup>: Matilde Cabrera González

#### Ejercicio 1.

Indicar los formatos de los archivos */etc/passwd*, */etc/group*, */etc/shadow* y */etc/gshadow*.

Formato de */etc/passwd*:

```
mati:x:1000:1000:mati,,,:/home/mati:/bin/bash
```

<nombre>:<password>:<uid>:<gid>:<descripción opcional><carpeta>:<Shell>

<nombre>: nombre de usuario

<password>: contraseña usuario

<uid>: id de usuario

<gid>: id de grupo (grupo al que pertenece el usuario)

<descripción opcional><carpeta>: home del usuario

<Shell>: Shell activo para el usuario

Formato */etc/group*:

```
mati:x:1000:
```

<grupo>:<x>:<GID>

<grupo>: nombre del grupo

<x>: contraseña del grupo, se encuentra cifrada en */etc/shadow*

<GID>: id de grupo (el cero se reserva para el grupo root)

Formato */etc/shadow*:

```
mati:$6$v7X1/Eyq$S38PdX3Ph9IicNYDert0xWJb3qddqeZAS1JMUj1uL5T9yNqMI/LFGv.5wkkLvBTASso82Uf1fNPJSMMy9uy38/:17801:0:99999:7:::
```

<nombre>:<password cifrado>:<1>:<2>:<3>:<4>:<5>:<6>

<nombre>: nombre de usuario

<password cifrado>: contraseña del usuario cifrada

<1>: días transcurridos desde 01/01/1970 donde el password fue cambiado por última vez.

<2>: número mínimo de días entre cambios de contraseña.

<3>: tiempo máximo en días de validez para la cuenta.

<4>: cuantos días antes de caducar la contraseña te avisa.

<5>: después de que la contraseña caduque, cuantos días tardará en deshabilitar la cuenta

<6>: fecha de caducidad, días desde 01/01/1970, donde la cuenta es deshabilitada y el usuario no podrá iniciar sesión.

Formato `/etc/gshadow`:

```
mati:!::
```

<grupo>: <password>:<admin>:<lista user>  
<grupo>: nombre del grupo  
<password>: contraseña del grupo  
<admin>: administradores del grupo  
<lista user>: lista de usuarios del grupo, separados por una coma.

## Ejercicio 2.

Modificar el archivo `/etc/login.defs` para que los usuarios creados a partir de ese momento tengan un valor asignado para las directivas `LOGIN_TIMEOUT`. Crear un usuario y comprobar que tiene efecto la citada directiva.

La directiva `LOGIN_TIMEOUT` viene predefinida a 60, lo dejamos a 2

```
#  
LOGIN_TIMEOUT          2
```

creamos un usuario nuevo “`sudo useradd nuevo`”

```
mati@mati-VirtualBox:~$ sudo useradd nuevo  
mati@mati-VirtualBox:~$ cat /etc/passwd  
nuevo:x:1001:1001::/home/nuevo:/bin/sh
```

Para ponerle password al usuario nuevo “`sudo passwd nuevo`”

Antes de seguir vamos a cambiar el intérprete de sh a bash.

```
mati@mati-VirtualBox:~$ sudo chsh -s /bin/bash nuevo  
[sudo] contraseña para mati:  
mati@mati-VirtualBox:~$ su nuevo  
Contraseña:  
nuevo@mati-VirtualBox:/home/mati$ cat /etc/passwd  
nuevo:x:1001:1001::/home/nuevo:/bin/bash
```

Hemos cambiado al usuario nuevo y nos ha dejado sin problemas. Hemos comprobado los logs y no se refleja que tenga ningún error.

## Ejercicio 3.

Crear un ACL para un archivo de vuestro sistema de forma que el usuario creado en el Ejercicio 2 tenga acceso de lectura y escritura.

Comprobamos si tenemos instalado attr, no es así, lo instalamos con “`sudo apt install attr`”. Ahora creamos un archivo llamado “`archivoprueba.txt`”

```

mati@mati-VirtualBox:~$ chmod archivoprueba.txt
chmod: falta un operando después de «archivoprueba.txt»
Pruebe 'chmod --help' para más información.
mati@mati-VirtualBox:~$ touch archivoprueba
mati@mati-VirtualBox:~$ ls
archivoprueba  Documentos  Imágenes  Plantillas  Vídeos
Descargas      Escritorio  Música    Público
mati@mati-VirtualBox:~$

```

Comprobamos los permisos por defecto con getfacl:

```

mati@mati-VirtualBox:~$ getfacl archivoprueba
# file: archivoprueba
# owner: mati
# group: mati
user::rw-
group::r--
other::r--

```

Creamos un ACL para “archivoprueba” de forma que el usuario “nuevo” tenga acceso de lectura y escritura:

```

mati@mati-VirtualBox:~$ setfacl -m u:nuevo:rw archivoprueba
mati@mati-VirtualBox:~$ getfacl archivoprueba
# file: archivoprueba
# owner: mati
# group: mati
user::rw-
user:nuevo:rw-
group::r--
mask::rw-
other::r--

```

Como vemos el archivo “archivoprueba” sigue perteneciendo al usuario y grupo de “mati”, pero el usuario “nuevo” tiene permiso de escritura y lectura tal como se pedía en el ejercicio.

#### Ejercicio 4.

En el sistema que tenemos en uso, indicar los archivos de configuración existentes y comentar la misión de un par de ellos y cómo lo hacen.

El directorio /etc/pam.d contiene un archivo de configuración por cada aplicación que solicita autenticación PAM, estos son los archivos existentes:

```

mati@mati-VirtualBox:~$ ls /etc/pam.d/
chfn          common-session-noninteractive  login          runuser-l
chpasswd      cron                          newusers      su
chsh          cups                          other          sudo
common-account  gdm-autologin                passwd        systemd-user
common-auth     gdm-fingerprint              polkit-1
common-password gdm-launch-environment       ppp
common-session  gdm-password                 runuser

```

**Runuser** permite ejecutar comandos con usuario sustituto e ID de grupo. Si no se da la opción -u, se ejecuta el respaldo a su semántica y shell compatibles. La diferencia entre los comandos runuser y su es que runuser no solicita una contraseña, ya que sólo puede ser ejecutado por root.

```
mati@mati-VirtualBox:~$ cat /etc/pam.d/runuser
#%PAM-1.0
auth            sufficient      pam_rootok.so
session         optional       pam_keyinit.so revoke
session         required       pam_limits.so
session         required       pam_unix.so
```

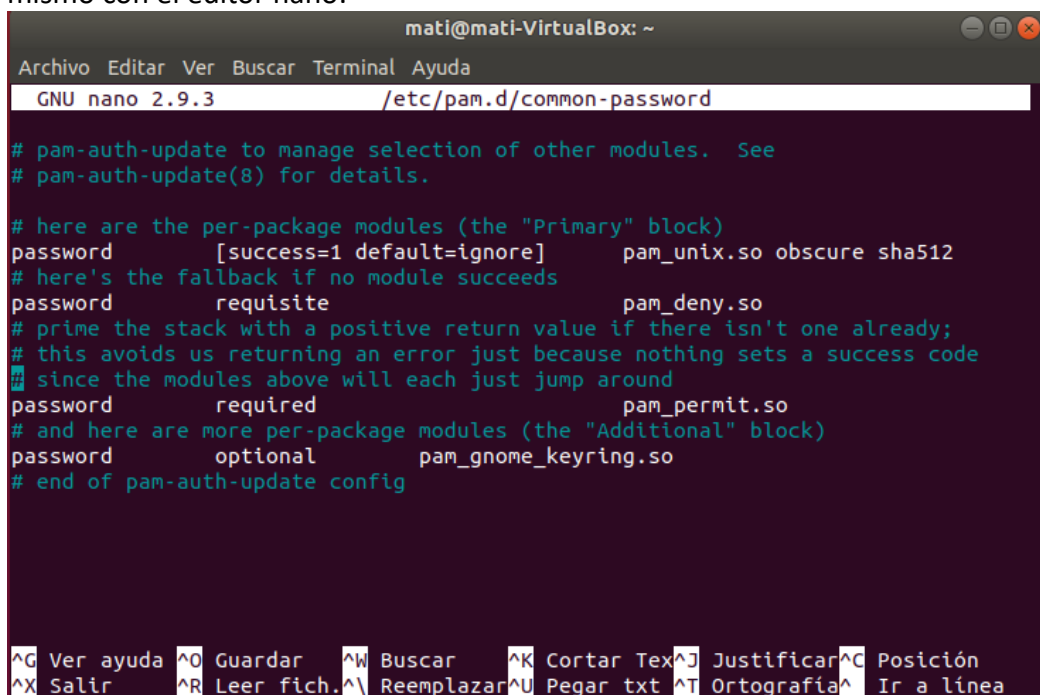
**Ppp** es una conexión directa entre dos pcs, ppp es un protocolo a nivel de enlace de datos, estableciendo una conexión directa entre dos nodos de una misma red. La conexión ppp no pasa por el cortafuegos, así que hay que tener especial cuidado, puede ser una puerta de entrada directa.

```
mati@mati-VirtualBox:~$ cat /etc/pam.d/ppp
#%PAM-1.0
# Information for the PPPD process with the 'login' option.
auth      required      pam_nologin.so
@include common-auth
@include common-account
@include common-session
mati@mati-VirtualBox:~$
```

## Ejercicio 5.

(a) Modificar la configuración para que se la autenticación exija que la clave de un usuario tenga una longitud mínima. Debemos utilizar el módulo *pam\_cracklib* ¡Cuidado! pues modificaciones inadecuadas pueden dejar sin acceso a usuarios que existen en el sistema.

La configuración para la autenticación está en “/etc/pam.d/common-password”, accedemos al mismo con el editor nano:



```
mati@mati-VirtualBox: ~
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
GNU nano 2.9.3 /etc/pam.d/common-password

# pam-auth-update to manage selection of other modules.  See
# pam-auth-update(8) for details.

# here are the per-package modules (the "Primary" block)
password      [success=1 default=ignore]      pam_unix.so obscure sha512
# here's the fallback if no module succeeds
password      requisite                       pam_deny.so
# prime the stack with a positive return value if there isn't one already;
# this avoids us returning an error just because nothing sets a success code
# since the modules above will each just jump around
password      required                       pam_permit.so
# and here are more per-package modules (the "Additional" block)
password      optional                       pam_gnome_keyring.so
# end of pam-auth-update config

^G Ver ayuda ^O Guardar ^W Buscar ^K Cortar Tex ^J Justificar ^C Posición
^X Salir ^R Leer fich. ^A Reemplazar ^U Pegar txt ^T Ortografía ^_ Ir a línea
```

Cambiamos para que la longitud mínima sea 4, mis contraseñas suelen tener 4 caracteres de

longitud.

```
# here are the per-package modules (the "Primary" block)
password      [success=1 default=ignore]      pam_unix.so obscure sha512 minlen=4
# here's the fallback if no module succeeds
password      requisite                        pam_deny.so
# prime the stack with a positive return value if there isn't one already;
# this avoids us returning an error just because nothing sets a success code
# since the modules above will each just jump around
password      required                        pam_permit.so
# and here are more per-package modules (the "Additional" block)
password      optional                        pam_gnome_keyring.so
# end of pam-auth-update config
```

**(b) Piensa otra modificación de tu preferencia e impleméntala. Por ejemplo, deshabilitar el acceso a root directo por consola, evitar que un usuario que no es el root tire el sistema, etc.**

Vamos a intentar modificar para que solo ciertos usuarios tengan acceso a nuestra maquina por conexión ssh, para ello primero cambiaremos el módulo pam\_access:

```
GNU nano 2.9.3 /etc/security/access.conf
# Archivos
# User "root" should get access from network 192.168.201.
# This term will be evaluated by string matching.
# comment: It might be better to use network/netmask instead.
#       The same is 192.168.201.0/24 or 192.168.201.0/255.255.255.0
#+ : root : 192.168.201.
#
# User "root" should be able to have access from domain.
# Uses string matching also.
#+ : root : .foo.bar.org
#
# User "root" should be denied to get access from all other sources.
#- : root : ALL
#
# User "foo" and members of netgroup "nis_group" should be
# allowed to get access from all sources.
# This will only work if netgroup service is available.
#+ : @nis_group foo : ALL
#
# User "john" should get access from ipv4 net/mask
#+ : john : 127.0.0.0/24
#
# User "john" should get access from ipv4 as ipv6 net/mask
#+ : john : ::ffff:127.0.0.0/127
#
# User "john" should get access from ipv6 host address
#+ : john : 2001:4ca0:0:101::1
#
# User "john" should get access from ipv6 host address (same as above)
#+ : john : 2001:4ca0:0:101:0:0:0:1
#
# User "john" should get access from ipv6 net/mask
#+ : john : 2001:4ca0:0:101::/64
#
# All other users should be denied to get access from all sources.
#- : ALL : ALL
```

Está todo comentado, hacemos nuestra propia regla. Al final del archivo añadimos lo siguiente:

```
#
# All other users should be denied to get access from all sources.
#- : ALL : ALL

+ : mati : ALL
+ : root : ALL
- : ALL : ALL
```

Ahora vamos a configurar el fichero de PAM para el servicio sshd, quedaría de la siguiente forma:

```
GNU nano 2.9.3 /etc/pam.d/sshd

# PAM configuration for the Secure Shell service

# Standard Un*x authentication.
@include common-auth

# Disallow non-root logins when /etc/nologin exists.
#account required pam_nologin.so //línea comentada
account required pam_access.so accessfile=/etc/security/access.conf

# Uncomment and edit /etc/security/access.conf if you need to set complex
# access limits that are hard to express in sshd_config.
# account required pam_access.so

# Standard Un*x authorization.
@include common-account

# SSU now needs to be the first rule in the rule. This ensures that you
```

Hemos añadido “account required pam\_access.so accessfile=/etc/security/access.conf”

## Ejercicio 6.

**Crear en el sistema un usuario con las características que deseéis, entrando como ese usuario cambiar la contraseña y analizar los archivos log para ver el mensaje correspondiente.**

Vamos a usar el usuario “nuevo” creado en el ejercicio 2. Cambiamos la contraseña, como estamos con usuario “mati”, primero tengo que introducir la contraseña de este usuario y después puedo cambiar sin problema la contraseña del anterior.

```
Oct  3 20:40:44 mati-VirtualBox passwd[2414]: pam_unix(passwd:chauthtok): password changed for nuevo
Oct  3 20:40:44 mati-VirtualBox passwd[2414]: gkr-pam: couldn't update the login keyring password: no old password was entered
Oct  3 20:40:44 mati-VirtualBox sudo: pam_unix(sudo:session): session closed for user root
```

Usando Ubuntu 18 no existe `/var/log/messages`, hemos encontrado dicho archivo en `/var/log/auth.log`

```
passwd: contraseña actualizada correctamente
mati@mati-VirtualBox:~$ cat /var/log/auth.log
Oct  1 09:50:33 mati-VirtualBox pkexec: pam_unix(polkit
ed for user root by (uid=1000)
```

### Ejercicio 7.

Modificar el archivo *sudoers* para que un usuario determinado tenga acceso a todas las órdenes del root.

```
Archivo Editar Ver Buscar Terminar Ayuda
GNU nano 2.9.3 /etc/sudoers Modi

# Members of the admin group may gain root privileges
%admin ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo  ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:

#includedir /etc/sudoers.d

#incluyo usuario nuevo para que tenga acceso de todas las ordenes root
nuevo ALL=(ALL:ALL) ALL
```

Y ahora comprobamos que nuevo tiene privilegios de root. Para ello intentamos algo que solo tenga permiso el root. Como el acceso a modificar este mismo archivo.

```
mati@mati-VirtualBox:~$ su nuevo
Contraseña:
nuevo@mati-VirtualBox:/home/mati$ cat /etc/sudoers
cat: /etc/sudoers: Permiso denegado
nuevo@mati-VirtualBox:/home/mati$ sudo cat /etc/sudoers
[sudo] contraseña para nuevo:
#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
```

### Ejercicio 8.

Analiza el contenido de estos archivos de registro del sistema de prácticas y comprueba que efectivamente se registran los eventos indicados.

Linux registra los eventos del sistema en el directorio */var/log*, vamos a ver que tenemos.



```

mati@mati-VirtualBox:~$ ls /var/log
alternatives.log      cups                  hp                   syslog.1
alternatives.log.1    dist-upgrade          installer             syslog.2.gz
apt                   dpkg.log              journal              syslog.3.gz
auth.log              dpkg.log.1            kern.log              tallylog
auth.log.1            faillog               kern.log.1            unattended-upgrades
bootstrap.log         fontconfig.log        lastlog              wtmp
btmp                  gdm3                  speech-dispatcher    wtmp.1
btmp.1                gpu-manager.log       syslog
mati@mati-VirtualBox:~$

```

Analizamos los indicadores en prácticas, lastlog nos indica el ultimo acceso hecho por los usuarios:

```

mati@mati-VirtualBox:~$ lastlog
Nombre          Puerto  De          Último
root            0       0           **Nunca ha accedido**
daemon          0       0           **Nunca ha accedido**
bin             0       0           **Nunca ha accedido**
sys             0       0           **Nunca ha accedido**
sync            0       0           **Nunca ha accedido**
games           0       0           **Nunca ha accedido**
man             0       0           **Nunca ha accedido**
lp              0       0           **Nunca ha accedido**
mail            0       0           **Nunca ha accedido**
news            0       0           **Nunca ha accedido**
uucp            0       0           **Nunca ha accedido**
proxy           0       0           **Nunca ha accedido**
www-data        0       0           **Nunca ha accedido**

```

Lista de todos los usuarios que han hecho login y logout desde que se creo el archivo

```

mati@mati-VirtualBox:~$ last
mati      :0      :0      Wed Oct 3 12:51   gone - no logout
reboot    system boot  4.15.0-34-generi Wed Oct 3 12:50   still running
mati      :0      :0      Tue Oct 2 23:17 - 08:46 (09:29)
reboot    system boot  4.15.0-34-generi Tue Oct 2 23:16 - 08:46 (09:30)
mati      :0      :0      Mon Oct 1 18:10 - 22:38 (1+04:28)
reboot    system boot  4.15.0-34-generi Mon Oct 1 18:10 - 22:39 (1+04:28)

wtmp empieza Mon Oct 1 09:52:51 2018
mati@mati-VirtualBox:~$

```

Los usuarios que aún están conectados con who

```

mati@mati-VirtualBox:~$ who
mati      :0      2018-10-03 12:51 (:0)
mati@mati-VirtualBox:~$

```

Todos los intentos fallidos de conexión de los usuarios del sistema.



```
mati@mati-VirtualBox:~$ sudo lastb  
btmp empieza Mon Oct  1 09:49:12 2018  
mati@mati-VirtualBox:~$
```

Sudo y messages no los tenemos en nuestro sistema operativo, encontramos la misma información en auth.log

Vamos a ver que realmente a registrado nuestros pasos:

```
Oct  3 21:20:34 mati-VirtualBox sudo:      mati : TTY=pts/0 ; PWD=/home/mati ; USER=root ; COMMAND=/bin/cat /var/log/lastlog  
Oct  3 21:20:34 mati-VirtualBox sudo: pam_unix(sudo:session): session opened for user root by (uid=0)  
Oct  3 21:20:34 mati-VirtualBox sudo: pam_unix(sudo:session): session closed for user root  
Oct  3 21:22:04 mati-VirtualBox sudo:      mati : TTY=pts/0 ; PWD=/home/mati ; USER=root ; COMMAND=/usr/bin/lastb  
Oct  3 21:22:04 mati-VirtualBox sudo: pam_unix(sudo:session): session opened for user root by (uid=0)  
Oct  3 21:22:04 mati-VirtualBox sudo: pam_unix(sudo:session): session closed for user root  
Oct  3 21:27:25 mati-VirtualBox sudo:      mati : TTY=pts/0 ; PWD=/home/mati ; USER=root ; COMMAND=/usr/bin/lastb  
Oct  3 21:27:25 mati-VirtualBox sudo: pam_unix(sudo:session): session opened for user root by (uid=0)  
Oct  3 21:27:25 mati-VirtualBox sudo: pam_unix(sudo:session): session closed for user root
```

## Ejercicio 9.

**Analizar las conexiones al sistema de prácticas y al de casa. ¿hay o ha habido alguna conexión ajena al equipo?**

En casa ya lo hemos visto en el ejercicio anterior. Lo repetimos en otra maquina y nos da la misma salida.