

# SSO Práctica 1

Andrés Merlo Trujillo

## Índice

<b>1. Primer ejercicio</b>	<b>1</b>
1.1. /etc/passwd . . . . .	1
1.2. /etc/group . . . . .	2
1.3. /etc/shadow . . . . .	2
1.4. /etc/gshadow . . . . .	3
<b>2. Segundo ejercicio</b>	<b>3</b>
<b>3. Tercer ejercicio</b>	<b>3</b>
<b>4. Ejercicio 4</b>	<b>4</b>
4.1. /etc/pam.d/chfn . . . . .	4
4.2. /etc/pam.d/chsh . . . . .	4
<b>5. Quinto ejercicio</b>	<b>4</b>
5.1. Apartado a . . . . .	4
5.2. Apartado b . . . . .	4
<b>6. Sexto ejercicio</b>	<b>5</b>
<b>7. Séptimo ejercicio</b>	<b>5</b>

## 1. Primer ejercicio

A continuación, voy a explicar el formato y el significado de cada uno de los campos. Para ello, voy a dividir cada archivo en subsecciones:

### 1.1. /etc/passwd

Este fichero está formado por líneas de 7 campos separados por “:”. Los campos y sus significados son los siguientes:

1. **Nombre de login:** Nombre de usuario.
2. **Contraseña encriptada opcional:** Contraseña encriptada del usuario. Si este campo tiene la letra “x” minúscula, significa que la contraseña se almacena en “/etc/shadow”.  
Si se encuentra vacío, significa que no hace falta contraseña para autenticar.  
Si comienza en exclamación, significa que la contraseña ha sido bloqueada.  
Además, si contiene una exclamación o un asterisco, significa que el usuario no podrá usar la contraseña para iniciar sesión (pero puede usar otro medio).
3. **User ID numerico:** ID del usuario.
4. **Group ID numerico:** ID del grupo al que pertenece.
5. **Nombre de usuario o campo de comentario:** Este campo sirve para poder poner un comentario sobre el usuario (por ejemplo: acción que realiza, para evitar confusión con dos usuarios similares, etc).
6. **Directorio home del usuario:** Directorio que será el home privado del usuario. Además sirve para poner la variable de entorno “\$HOME”

7. **Interprete opcional de comando de usuario:** Shell que usará el usuario por defecto (bash, sh, zsh, fish, etc). Además, pondra la variable de entorno “\$SHELL” a este valor.

## 1.2. /etc/group

Este fichero está formado por 4 campos separados por “:”. El significado de cada campo es el siguiente:

1. **Nombre del grupo:** Nombre del grupo. Este nombre debe ser único en el sistema.
2. **Contraseña:** Contraseña del grupo. Si es una letra “x” minúscula significa que la contraseña encriptada se encuentra en “/etc/gpasswd”.
3. **Group ID:** Indica el ID del grupo. Este valor debe ser único en el sistema.
4. **Usuarios:** Lista de usuarios separados por coma (“,”) los cuales son miembros del grupo.

## 1.3. /etc/shadow

Este fichero está formado por líneas de 9 campos separados por “:”. Los campos y sus significados son los siguientes:

1. **login name (nombre de login):** Nombre de la cuenta del usuario. Debe existir en el sistema.
2. **encrypted password (contraseña encriptada):** Contraseña encriptada del usuario especificado en “login name”. Si este campo está vacío, significa que ese usuario no requiere contraseña para iniciar sesión.

Además, en caso de que la contraseña comience con una exclamación (“!”), significa que la contraseña ha sido bloqueada.

Por último, si la contraseña contiene el carácter de exclamación mencionado anteriormente o asterisco (“\*”), significa que no puede iniciar sesión (si es exclamación también se cumple lo de arriba).

3. **date of last password change (fecha del ultimo cambio de contraseña):** El último cambio de contraseña, expresado como el número de días desde el epoch (1 de enero de 1970).

Además, si el valor es 0 significa que el usuario debe cambiar la contraseña en el próximo login.

En cambio, si el campo está vacío significa que las contraseñas no tienen edad (y por tanto no se cumplen estas restricciones).

4. **minimum password age (edad mínima de la contraseña):** Número de días que el usuario tiene que esperar antes de poder cambiar la contraseña de nuevo. Un valor 0 ó vacío indica que no hay un mínimo de días.

5. **maximum password age (edad máxima de la contraseña):** Número máximo de días en los cuales la contraseña “caduca” (tiene que cambiarla). Al pasar este número de días, el sistema pedirá al usuario que cambie la contraseña.

Si el valor máximo es mayor que el del campo anterior, el usuario no podrá cambiar su contraseña.

Por último, si el campo está vacío, se deshabilitará este servicio junto con “password warning period” y “password inactivity period”.

6. **password warning period (periodo de advertencia de la contraseña):** El número de días antes de que la contraseña “caduque” durante los cuales se le advierte al usuario.

Un valor 0 o cadena vacía indica que no habrá advertencias.

7. **password inactivity period (periodo de inactividad de la contraseña):** Número de días después de que la contraseña haya “caducado” en el cual debería ser aceptada. Al pasar este periodo, el usuario no podrá iniciar sesión.

Un campo vacío indica que no se cumple esta regla.

8. **account expiration date (fecha de expiracion de la cuenta):** La fecha en la que la cuenta expira. Esta fecha se expresa como el numero de dias desde el epoch.

La diferencia con la expiracion de una contraseña es que, si la cuenta expira, no podra iniciar sesion de ninguna forma, mientras que si la contraseña expira, tendrá otros medios para iniciar sesion.

El campo vacio indica que la cuenta no expira. Ademas, no se debe usar el valor 0 ya que se puede interpretar como que la cuenta expira en el epoch o que no expira.

9. **reserved field (campo reservado):** Este campo está reservado para usos futuros.

#### 1.4. /etc/gshadow

Este fichero está tambien formado por 4 campos separados por el símbolo “:”. El significado de cada campo es el siguiente:

1. **Nombre del grupo:** Nombre del grupo. Debe existir en el sistema.
2. **Contraseña encriptada:** Contraseña encriptada que sirve para que un usuario que no es miembro del grupo obtenga los permisos.  
Si el campo está vacio, entonces cualquier usuario puede obtener los privilegios del grupo.  
Si la contraseña comienza por una exclamacion, significa que esta está bloqueada.  
Si contiene una exclamacion o asterisco, los usuarios no podran acceder al grupo si no estan en el.
3. **Administradores:** Lista de usuarios separados por coma que puede realizar operaciones como cambiar la contraseña del grupo o administrar los usuarios del mismo.
4. **Miembros:** Lista de usuarios separados por coma. Los miembros del grupo pueden acceder al mismo sin necesitar la contraseña.

## 2. Segundo ejercicio

En este ejercicio se pide modificar el valor de la variable “LOGIN\_TIMEOUT” y comprobar sus efectos con un usuario nuevo que se haya creado manualmente.

Para ello, modifiko la variable, que estaba por defecto a 60 segundos:

Y lo cambio a otro valor, por ejemplo, 5 segundos:

A continuacion, creo el usuario llamado “prueba”, le cambio la contraseña y hago login con él desde la terminal. Cuando se encuentre en la parte de pedir la contraseña de este usuario nuevo, se espera un tiempo hasta que la terminal devuelva un mensaje:

Como se puede ver, pone que han pasado 5 segundos y el acceso ha caducado.

Ahora, pruebo con otro valor, por ejemplo **PONER AQUI EL TIEMPO QUE QUERIA** segundos:

E intento iniciar sesion de nuevo con el usuario “prueba” y espero en la parte de la contraseña.

Como se puede ver, el timeout ahora es distinto.

## 3. Tercer ejercicio

En este ejercicio se pide crear un archivo y darle, mediante un ACL, permisos de lectura y escritura al usuario creado (en mi caso sigue siendo “prueba”).

Para ello, mediante la orden “touch” creo el archivo denominado “ejercicio3”.

Ahora bien, al menos en Ubuntu 22.04 no están las ordenes “getacl/setacl”, sino que se llaman “getfacl/setfacl”. El resultado no varía y tienen las mismas sintaxis.

Ahora, con la orden “setfacl” se le dará al usuario “prueba” permisos “rw”.

Y ahora mostramos con “getfacl” el archivo anterior:

Como se puede observar, ahora aparece una linea que indica que el usuario “prueba” tiene permisos “rw”.

## 4. Ejercicio 4

Con el comando “ls” muestro los archivos que se encuentran en el directorio “/etc/pam.d”:  
A continuacion explicare dos archivos:

### 4.1. /etc/pam.d/chfn

Permite cambiar la informacion personal de un usuario tales como: el nombre, el numero de telefono, de habitacion, etc. Estos datos luego pueden ser leidos por comandos como “finger”.

El contenido del archivo es:

La funcion de la linea 7 es para no pedir la contraseña al usuario root cuando esté usando este comando. Para ello, hace uso del campo de control sufficient, que hace que si tiene exito retorne sin ejecutar mas modulos. Ademas, hgace uso del modulo “pam\_rootok.so” que hace que solo tenga extio si el usuario tiene el UID a 0 (es el usuario root).

### 4.2. /etc/pam.d/chsh

El comando “chsh” permite cambiar la shell por defecto del usuario que lo invoca. Si no se le pasa ningun parametro se activa el modo interactivo para realizar el cambio de shell.

El contenido del archivo es el siguiente:

Como se puede ver en la linea 8, esta llamada lo que hace es prohibir el cambio de shell a no ser que se encuentre listada en “/etc/shells”. Esto se consigue mediante el campo de control “required”, que provocará un fallo de autenticacion en el sistema (ejecutará la linea siguiente, pero al ser irrelevante, no pasa nada) si el modulo falla. Tambien se consguie mediante la llamada al modulo “pam\_shells.so”, que hace que si la shell pasada como parametro no se encuentra en “/etc/shells” dé un fallo.

La funcion de la linea 12 es de permitir al superusuario cambiar la shell sin ser necesario introducir la contraseña. Esto se realiza mediante el campo de control “sufficient” y el modulo “pam\_rootok.so”. Con “sufficient”, cuando la orden tiene exito retorna sin ejecutar los demas modulos **COMPROBAR AFIRMACION: (Como es el ultimo puede retornar sin prblema)**. Ademas, con el modulo “pam\_rootok.so” autoriza al usuario con el UID 0 (root).

## 5. Quinto ejercicio

### 5.1. Apartado a

Es necesario modificar el archivo PAM “common-password” y en Ubuntu 22.04 ya como primera linea aparece el uso del modulo “pam\_pwquality”.

Ahora bien si leemos el manual de este modulo con “man 8 pam\_pwquality” se puede ver que hay un argumento denominado “minlen” y que valor por defecto es 8. No obstante, no se puede bajar del valor 4, ya que es un limite que tiene “Cracklib” y mostrara que la contraseñae es muy corta. Por eso, voy a poner el limite a **15 caracteres**.

Y ahora al usar el comando “passwd” y poner una contraseña con menos de 15 palabras, muestra un error:

Y al agotarse los intentos (que son 3) se sale del programa.

### 5.2. Apartado b

En esta parte he restringido el acceso al comando “su” para así evitar que un usuario que ponga el comando sin “sudo” pueda entrar. Si ponen “sudo su” sí van a poder entrar, pero esto es así ya que son usuarios administradores (y es una decisión de diseño, ya que en otro caso no podría usar nadie “sudo”), en ese caso lo recomendable es deshabilitar el acceso al grupo “sudo” (en el caso de Ubuntu) para que no lo pueda usar (editando el archivo sudoers mediante el comando “visudo”).

Para conseguir esto, es neceasrio modificar el arcvhio “/etc/pam.d/su” y añadir la siguiente línea al principio:

Ahora, al ejecutar el comando “su” con un usuario nomral aparece lo siguiente:

En cambio, si el usuario puede usar “sudo”, si puede acceder.

La línea que he añadido lo que hace es comprueba que la cuenta sea root (UID=0) y en caso de no serlo, no sigue ejecutando el archivo provocando un error de autenticación.

## 6. Sexto ejercicio

En Ubuntu 22.04 los cambios de contraseña no se almacenan en “/var/log/messages”, sino en “/var/log/auth.log”. [Enlace](#) a la guía.

Voy a crear el usuario “ejercicio6” y le voy a cambiar la contraseña:

Y ahora, al mostrar el archivo “/var/log/auth.log” aparecen las siguientes líneas:

## 7. Séptimo ejercicio