1.- Gestión de usuarios.

Una de las responsabilidades del administrador de sistemas es gestionar los usuarios. Esto consiste en:

- Dar de alta y baja a los usuarios.
- Administrar las contraseñas.
- Configurar las cuentas de usuarios.
- Administrar las plantillas de usuarios.
- Gestionar los grupos de usuarios.

1.1.- Archivos de configuración de usuarios.

Estos archivos son:

- /etc/passwd: archivo que contiene la información de los usuarios, excepto las contraseñas.

- /etc/shadow: archivo que contiene información sobre las contraseñas de los usuarios.

Archivo /etc/passwd:

En este archivo contiene las cuentas de:

- Usuarios que pueden acceder al sistema y creadas por el root.
- Aplicaciones y demonios creadas para ser utilizadas por los diferentes servicios que ofrecen.

Este archivo contiene una serie de líneas con los campos separados por: (dos puntos).

Login:password:UID:GID:Descripcion:Directorio_personal:Shell

- *Login*: nombre de la cuenta. Identificador de usuario para acceder al sistema.
- *Password*: contraseña de usuario. En la contraseña aparece una “x” siempre.
- *UID*: numero identificador de usuario. No se puede cambiar. Las cuentas de aplicaciones, demonios y el root tienen los números más bajos. Los números de cuentas empiezan a partir del número que parece definido en el archivo /etc/login.defs en el parámetro UID_MIN.
- *GID*: numero identificador de grupo al que pertenece, en el inicio de sesión, el usuario. Es un número único para cada grupo. Varios usuarios pueden pertenecer a un mismo grupo. Toda la información de los grupos se almacena en /etc/group. Al crear una cuenta de usuario si no se especifica ningún nombre de grupo, se crea un grupo con el mismo nombre que la cuenta de usuario.
- *Descripción*: Datos que describen al usuario. Es información variada aunque sigue una serie de normas. El comando finger muestra la información de este campo.
- *Directorio personal*: ruta completa o absoluta del directorio de trabajo del usuario.

- *Shell*: interprete de comandos que ejecuta el servidor. Si aparece el Shell /sbin/nlogin indica que no tiene permisos para conectarse al sistema, se trata de un usuario creado para un servicio.

Archivo /etc/shadow

Archivo que almacena información sobre las contraseñas. Incluye tanto las contraseñas cifradas correspondientes a la cuentas del archivo /etc/passwd como los datos sobre el vencimiento de la contraseña y bloqueo de la cuenta.

Hay dos conceptos a tener en cuenta en este apartado:

- **Vencimiento de la contraseña**: tiempo de vida de una contraseña desde el último cambio. Pasado este tiempo, el sistema obliga al usuario a cambiar la contraseña, tras haberle ofertado un tiempo de prórroga para realizar el cambio.
- **Bloque de cuenta**: la cuenta se bloquea, impidiendo el acceso a la misma, pasado el tiempo establecido sin cambiar la contraseña.

Cada línea del archivo contiene los siguientes campos:

Login:contraseña:utlmod:min:max:aviso:inactivo:expira

- *Login*: nombre de la cuenta.
- *Contraseña*: puede contener la siguiente información:
  - Una contraseña cifrada.
  - Un asterisco (*) indica que la cuenta tiene la contraseña deshabilitada.
  - Una admiración (!) indica que la cuenta está bloqueada y no puede usarse.
  - Dos admiraciones (!!) indica que no está definida.
- *Ult mod*: tiempo que ha transcurrido desde el último cambio de la contraseña.
- *Min*: número de días que transcurren desde que la contraseña se cambia hasta que puede volver a ser modificada por el usuario.
- *Max*: vencimiento de la contraseña. Número máximo de días que pueden pasar desde el último cambio de contraseña hasta que el sistema obligue al usuario a cambiarla de nuevo.
- *Aviso*: número de días previos al vencimiento de la contraseña (Max) en los que se le sugiere que la cambie ya que esta próxima la expiración.
- *Inactivo*: número de días que transcurren entre el vencimiento y el bloqueo de la cuenta. Si estos días expiran, el usuario pasar a estar inactivo y no tener acceso a la cuenta.
- *Expira*: fecha en la que la cuenta se deshabilita.

1.2.- Gestión de usuarios con la suite Shadow.

La suite Shadow es un conjunto de comandos y herramientas que permiten mejorar la seguridad en la gestión de los usuarios y ocultar las contraseñas.

- Crear un usuario

useradd <opciones> login

Si no se especifica ninguna opción, se asignaran los valores por defecto.

Opción	Descripción
-m	Crea también el directorio personal. Se incluye a veces por defecto.
-u	Especifica el UID numérico, para forzarlo. Si no se asigna por defecto siguiendo la reglas del fichero /etc/login.defs
-g	Especifica el grupo primario del usuario por GID o por su nombre.
-G	Especifica los grupos adicionales separados por comas.
-d	Ruta del directorio personal. En general, /home/login pero no se puede especificar cualquier ruta.
-c	Un comentario asociado a la cuenta.
-s	Shell (interprete de comandos) por defecto del usuario
-p	La contraseña del usuario. La contraseña debe estar ya cifrada.

- Cambiar la contraseña

passwd <opciones> <login>

El comando passwd permite gestionar las contraseñas, pero también las autorizaciones de conexión, así como la mayoría de los campos presentes en /etc/shadow.

El usuario root tiene derecho a modificar las contraseñas de todos los usuarios del sistema, sin que sea preciso conocer la contraseña anterior.

Opción	Definición
-l	Lock: bloquea una cuenta al añadir una ! delante de la contraseña cifrada
-u	Unlock: desbloquea la cuenta. No es posible activar una cuenta cuando no tenga contraseña. Hay que utilizar -f para ello.
-d	(root) Suprime la contraseña de la cuenta.
-n <num>	(root) Duración de vida mínima en días de la contraseña.
-x <num>	(root) Duración de vida máxima en días de la contraseña.
-w <num>	(root) Numero de días antes de un aviso
-i <num>	(root) Periodo de gracia antes de la desactivación si ha vencido la contraseña.

- Eliminar un usuario.

userdel <opción> login

Suprime un usuario. Por defecto no se suprime el directorio personal. Para ello se utiliza la opción **-r**

- Modificar la información de un usuario.

usermod <opciones> login

Modifica la cuenta del usuario. Utiliza la misma sintaxis y opciones que useradd, pero dispone de unas opciones distintas.

Opción	Definición
-L	Bloqueo de la cuenta, como passwd -l
-U	Desbloqueo de la cuenta, como passwd -u
-e <n>	Vencimiento: la contraseña expira n días después del 01/01/1970
-u <UID>	Modifica el UID asociado al login. Se modifica en consecuencia el propietario de los ficheros que pertenecen al antiguo UID dentro del directorio personal.
-l <login>	Modifica el nombre de login.

- Modificar el Shell asignado.

chsh <opciones> <login>

Este comando permite al usuario modificar de manera definitiva el Shell de conexión. No se puede elegir cualquier cosa. El Shell debe estar presente en /etc/shells. Solo root tiene derecho a modificarla a otros usuarios.

Opción	Definición
-s Shell	Modifica el Shell indicado
-l	Lista los Shell existentes en el archivo /etc/shells

- Cambiar el comentario.

chfn

El usuario puede modificar el comentario del fichero /etc/passwd con este comando. Es preferible utilizarlo de manera interactiva.

- Cambiar la identidad

su <usuario>

El usuario puede adoptar la identidad de otra persona mientras dura un comando o toda una sesión. Se suele tratar de root.

El comando su permite abrir una sesión, o ejecutar un Shell, o un comando dado, con otra identidad. Obviamente, se debe conocer la contraseña de este usuario.

- Mostrar información de grupos de un usuario.

id

Imprime **UIDs** (Número de identificación de usuario) y **GIDs** (Número de identificación de grupo) reales y efectivos.

- Modificar la información sobre las contraseñas.

chage <opciones> <usuario>

Es un comando de root. Iniciado sin argumento, es interactivo. Pero también se pueden especificar opciones.

Opción	Definición
-m	Mindays: equivale a passwd -n.
-M	Maxdays: equivale a passwd -x.
-d	Fecha de última modificación de la contraseña (desde el 01/01/1970)
-E	Fecha de vencimiento de la contraseña (desde el 01/01/1970)
-I	Inactive: equivale a passwd -i
-W	Warndays: equivale a passwd -w
-l	List: muestra todos los detalles.

2.- Gestión de grupos.

Los grupos permiten a los administradores conceder permisos a un conjunto de usuarios simultáneamente.

En Linux, un usuario solo puede pertenecer a un grupo en un determinado momento. El grupo al que pertenece cuando se conecta al sistema es el que aparece en el archivo /etc/passwd, pero a lo largo de su conexión puede cambiar de un grupo a otro para tener otros permisos.

2.1.- Archivos de configuración de grupo .

Archivo /etc/group

El archivo /etc/group contiene la definición de los grupos de usuarios, y para cada uno la lista de los usuarios de los cuales es el grupo secundario. Cada línea se compone de cuatro campos

Group:password:GID:user1,user2,…

- *Group*: nombre del grupo
- *Password*: la contraseña asociada al grupo.
- *GID*: identificador del grupo.
- *Lista de usuarios*: usuarios que forman parte de este grupo

Archivo /etc/gshadow

El fichero gshadow es el equivalente al fichero /etc/shadow pero para los grupos. Sin embargo, la mayoría de las distribuciones Linux no lo soportan por defecto.

2.2.- Gestión de grupos con la suite Shadow.

Los comandos de la suite Shadow son:

- Creación de un grupo

groupadd <opciones> grupo

Es un comando de root. Permite crear grupos.

Opción	Definición
-g	Valor numérico del identificador de grupo. Este valor debe ser único.

- Modificación de un grupo

groupmod <opciones> nuevo_nombre nombre_antiguo

Permite modificar información del grupo. Es comando de root.

Opción	Definición
-n <nombre>	Renombra el grupo
-g <GID>	Modifica el GID. No se modifica el grupo al que pertenecen los ficheros correspondientes
-A <user>	Añade el usuario especificado en el grupo (grupo secundario)
-R <user>	Suprime el usuario especificado del grupo

- Supresión de un grupo

groupdel nombre_grupo

Permite suprimir un grupo. Primero el comando comprueba si el grupo que se desea suprimir es el grupo primario. Si es así, no se permite suprimir el grupo.

- Listado de grupos

groups

Se usa para mostrar los grupos a los que pertenece un usuario.

- Logearse en un nuevo grupo

newgroup grupo

Se utiliza para cambiar el ID de grupo actual durante una sesión de inicio de sesión.

2.3.- Configuración avanzada

a) /etc/default/useradd

Este fichero contiene un cierto número de variables que definen las reglas por defecto que se deben aplicar en el momento de crear un usuario:

- su grupo.
- La raíz de su directorio personal.
- Si esta activo o no.
- El Shell
- Su grupo o sus grupos secundarios.
- El lugar donde se sitúa la estructura básica del directorio básico
- Etc.

b) /etc/login.defs

Muchos comandos, como login, useradd, groupadd, passwd utilizan el fichero para definir algunos valores por defecto y la validez de los logins. Su contenido puede variar en función de las distribuciones. Suele contener:

- Los UID mínimo y máximo durante la creación de un usuario.
- Los GID mínimo y máximo durante la creación de un grupo.
- Los comandos que hay que llamar para la creación/modificaciones/eliminaciones de un usuario.
- Las reglas por defecto para la validez de las contraseñas.
- La creación o no de un directorio personal.
- Etc.

c) /etc/issue

Cuando un usuario se conecta por consola, se suele utilizar un mensaje justo antes de la línea de comandos de inserción de su login. El fichero /etc/issue contiene este mensaje. Por defecto, suele contener el nombre de la distribución Linux y el número de versión del núcleo. Se trata de un mensaje de bienvenida y por este motivo puede contener todo lo que se desee. Admite secuencia de caracteres para poderle aplicar sustituciones.

d) /etc/issue.net

El mensaje de bienvenida puede ser diferente cuando un usuario se conecta desde una consola remota (telnet, ssh, etc.). A menudo es el mismo que /etc/issue, pero sin los caracteres de control relacionados con un Shell dado.

e) /etc/motd

Motd significa Message of the day, el mensaje del día. Una vez conectado un usuario desde una consola (local o remota), se puede visualizar un mensaje. El administrador puede modificarlo editando el fichero. Por defecto esta vacio. Se puede modificar para, por ejemplo, avisar a los usuarios de un reinicio de mantenimiento en un día y hora determinados, etc...

3.-Gestion de permisos de archivos y directorios.

3.1.- Permisos básicos.

El papel de un sistema operativo es también el de asegurar la integridad y el acceso a los datos, lo que es posible gracias a un sistema de permisos. A cada fichero o directorio se le asignan unos privilegios que le son propios, así como autorizaciones de acceso individuales. En el momento intentar el acceso, el sistema comprueba si está autorizado.

Permiso	Significado
General	
r	Readable (lectura)
w	Writable (escritura)
x	Executable (ejecutable como programa)
Fichero normal	
r	Se puede leer el contenido del fichero, cargarlo en memoria, listarlo y copiarlo.
w	Se puede modificar el contenido del fichero. Se puede escribir dentro. Modificar el contenido no significa poder eliminar el fichero (ver permisos de directorios).
x	Se puede ejecutar el fichero desde la línea de comandos si se trata de un programa binario (compilado) o de un script (Shell, perl…)
Directorio	
r	Se pueden listar (leer) los elementos del directorio (catalogo). Sin esta autorización, ls y los criterios de filtro de un directorio y su contenido no serian posibles. No obstante, puede ser accediendo a un fichero si conoce su ruta de acceso.
w	Se pueden modificar los elementos del directorio (catalogo), y es posible crear, volver a nombrar y suprimir ficheros en este directorio. Es este permiso el que controla el permiso de eliminación de un fichero.
x	Se puede acceder al catalogo por CD y se puede listar. Sin esta autorización, es imposible acceder al directorio y actuar en su contenido, que pasa a estar cerrado.

Así, para un fichero:

rwx	**r-x**	**r--**
Permisos para el propietario de lectura, escritura y ejecución.	Permiso para los miembros del grupo de lectura y ejecución.	Permisos para el resto del mundo de lectura únicamente.

3.2.- Modificación de los permisos.

Cuando se crea, un fichero o un directorio se dispone de permisos por defecto. Utilizamos el comando **chmod** para modificar los permisos de un fichero o un directorio. Existen dos métodos para modificar estos permisos: mediante símbolos o mediante el sistema octal de representación de permisos. Solo el propietario de un fichero puede modificar sus permisos (mas el administrador del sistema). El parámetro **-R** cambia los permisos de forma recursiva.

a) Mediante símbolos

chmod modificaciones fich1 [fich2 ...]

Si hay que modificar permisos de:

- Propietario -- **u**
- Grupo -- **g**
- Resto -- **o**
- Todos -- **a**

Para **añadir** permisos, se utiliza el carácter **+**

Para **retirarlos** o **eliminar** permisos, se utiliza el carácter **–**

Para **no tener en cuenta los parámetros anteriores**, el carácter **=**

Finalmente, poner el permiso cuyos símbolos son: r,w,x

Ejemplo: chmod g+w fich1

b) Sistema octal

La sintaxis es idéntica a la de los símbolos. A cada permiso le corresponde un valor octal, posicional y acumulable. Para modificar los tres permisos (rwx), hacen falta tres bits: cada uno tomaria el valor 0 ó 1 según la presencia o no del derecho. 2^3= 8 de ahí la notación octal.

- r vale 4
- w vale 2
- x vale 1

La tabla siguiente nos ayudara:

Propietario			**Grupo**			**resto**		
r	**w**	**x**	**r**	**w**	**x**	**r**	**w**	**x**
400	200	100	40	20	10	4	2	1

La modificación octal de permisos no es sutil y no permite modificar un solo derecho. Es la totalidad de los permisos lo que se modifican de una sola vez.

3.3.- Máscara de permisos

a) Restringir unos permisos de manera automática.

En el momento de la creación de un fichero o directorio, se les asigna unos permisos automáticamente. Suele ser: rw-r--r--(644) ó rw-rw-rw-(666) para un fichero y rwxr-xr-x (755) para un directorio. Una máscara de permisos controla estos valores. Se puede modificar con el comando **umask**. El comando **umask** coge como parámetro un valor octal del cual cada permiso individual será suprimido de los derechos de acceso máximo del fichero o del directorio.

- Por defecto, se crean todos los ficheros con los permisos 666
- Por defecto, se crean todos los directorios con los permisos 777
- Luego se aplica la máscara.
- La máscara es la misma para el conjunto de los ficheros.
- Una máscara no modifica los permisos de los ficheros existentes, sino solamente los creados a partir de este momento.

El comando umask sin parámetro muestra la máscara por defecto.

b) Cálculo de la máscara

Para un fichero:

Predeterminado	rw- rw- rw-	666
Retirar	--- -w- -w-	022
Resta	rw- r-- r--	644

Para un directorio:

Predeterminado	rwx rwx rwx	777
Retirar	--- -w- -w-	022
Resta	rwx r-x r-x	755

Aplicar una máscara no es sustraer, sino suprimir permisos de los establecidos por defecto, permiso a permiso.

Ejemplo: Partiendo de una máscara 023, se quiere saber qué permisos tendrán por defecto los archivos y directorios del sistema.

Para ficheros:

Predeterminado	rw- rw- rw-	666
Retirar	--- -w- -wx	023
Resta	rw- r-- r--	644

Para directorios:

Predeterminado	rwx rwx rwx	777
Retirar	--- -w- -wx	023
Resta	rwx r-x r--	754

3.4.- Cambiar de propietario y grupo.

Es posible cambiar de propietario y grupo de un fichero con ayuda de dos comandos:

- **chown** (change owner) : permite cambiar el propietario del archivo o directorio, así como del grupo.

  chown nuevo_propietario fich1 [fich2 …]

  chown nuevo_propietario:nuevo_grupo fich1 [fich2….]

- **chgrp** (change group): permite cambiar el grupo al que pertenece el archivo.

  chgrp nuevo_grupo fich1 [fich2….]

  La opción **-R** cambia la propiedad de manera recursiva.

Para los dos comandos, no se modifican los permisos anteriores ni la ubicación del fichero. Solo root tiene el permiso de cambiar el propietario de un fichero. Pero un usuario puede cambiar el grupo de un fichero, si forma parte del nuevo grupo.

3.5.- Permisos especiales.

Existen tres permisos especiales en Linux:

a) **Bit SetUID (S)**: se usa para que un usuario ejecute un archivo con los permisos del propietario del archivo.
b) **Bit SetGUID (G)**: se usa para que usuario ejecute un archivo con los permisos del grupo propietario del archivo.
c) **Bit sticky (T)**: se usa en los directorios para permitir únicamente a los propietarios de los archivos contenidos en los directorios, renombrarlos o borrarlos.

El comando chmod permite ubicar SUID-Bit , GUID-Bit y Sticky bit.

- *chmod u+s comando (SUID-Bit)*
- *chmod g+s comando (GUID-Bit)*
- *chmod u+t comando (Sticky bit)*

Los valores en octal son 4000 para SUID-Bit , 2000 para GUID-Bit y 1000 para Sticky bit.

- *chmod 4000 comando*
- *chmod 2000 comando*
- *chmod 1000 comando*