Bibliografía

Sarwar, S. M.; Koretsky, R.; Sarwar, S. A. El libro de LINUX. Addison Wesley, 2005.

Jorba, J; Suppi, R. Administración avanzada de GNU/Linux. Fundació per a la Universitat Oberta de Catalunya , 2007 Sobell, Mark G. A Practical Guide to Ubuntu Linux. Prentice Hall, 2009

Gestionar usuarios y grupos en Ubuntu. http://www.ubuntu-guia.com/2009/09/gestion-de-usuarios-y-grupos-en-ubuntu.html Como utilizar los comandos su y sudo. http://www.ubuntu-guia.com/2012/08/comandos-su-y-sudo.html Manual de sudo, visudo y sudoers. http://www.linuxtotal.com.mx/?cont=info_admon_014

DJG's Sudo Guide. http://www.linuxhelp.net/guides/sudo/

1. GESTIÓN DE USUARIOS Y GRUPOS

El sistema Linux es un sistema operativo multiusuario. Para que múltiples usuarios puedan hacer uso del sistema de una forma segura y controlada, el sistema debe disponer de mecanismos de administración y seguridad para proteger los datos de cada usuario, así como para asegurar el correcto funcionamiento del sistema. Entre estos mecanismos cabe destacar las **cuentas de usuario** y los **grupos**.

En un sistema Linux, los usuarios disponen normalmente de una cuenta junto con espacio en disco para ubicar sus ficheros y directorios. Se pueden distinguir tres tipos diferentes de cuentas:

(a) Cuenta del administrador o superusuario (identificador de usuario *root*, UID=0): posee todos los permisos y acceso completo a la máquina y a los ficheros de configuración. Se recomienda usar esta cuenta sólo para realizar operaciones de administración ya que pone en riesgo el sistema al garantizar acceso privilegiado a cada programa en ejecución.

(b) Cuentas de usuario: son las cuentas que poseen los usuarios del equipo. Estas cuentas tienen los permisos restringidos al uso de los archivos que cuelgan de su directorio de inicio y a algunos directorios particulares (por ejemplo, /tmp), así como al uso de algunos dispositivos.

(c) Cuentas especiales de los servicios: son cuentas que no pueden iniciar sesión, pertenecen al sistema y tienen acceso a servicios específicos, con lo cual asumen distintos privilegios del superusuario. Por ejemplo: lp, proxy, mail, etc

Para poder administrar los permisos de los usuarios de una forma más flexible, el sistema Linux permite la organización de usuarios en *grupos* y establecer permisos a los grupos.

Todos los usuarios pertenecen al menos a un grupo que es el **grupo principal** del usuario, también llamado **grupo primario** del usuario, pero pueden pertenecer a más grupos. En caso de que pertenezcan a más grupos, éstos serán **grupos secundarios.** Los grupos pueden contener varios usuarios, pero nunca podrán contener a otros grupos.

La gestión de grupos y usuarios sólo la puede realizar el administrador. En Ubuntu existen dos formas de gestionarlos:

1. mediante comandos introducidos en la línea de comandos de un terminal, y 2.

mediante un modo gráfico.

La primera técnica es más potente ya que permite realizar varias acciones a la vez y es la que se va a describir, a grandes rasgos, en este guión.

La información de los grupos y de los usuarios que reconoce el sistema está incluida en los siguientes ficheros de texto:

/etc/passwd cada línea de este fichero proporciona información sobre un usuario. Su contenido podría ser:

nombre_usuario:x:identificador_usuario:identificador_grupo_principal:comentario:directorio_inicio: shell

/etc/shadow contiene las contraseñas encriptadas de los usuarios. Parte del contenido de este fichero sería:
nombre_usuario:palabra_encriptada

/etc/group cada línea contiene los parámetros y los usuarios de cada grupo, excepto para el grupo principal de cada usuario, el cual figura en el archivo */etc/passwd*. Un posible contenido sería:

nombre_grupo:contraseña_grupo:identificador_grupo:lista_usuarios (separados por coma) /etc/gshadow

contiene la contraseña asociada a cada grupo, aunque no es muy común.

En el directorio */etc/skel* se guardan los “esqueletos” que se copian en el directorio de inicio de los nuevos usuarios cuando se crean.

1.1. Gestión de Grupos

Entre las operaciones que se puede realizar con los grupos caben destacar las siguientes:

A) Añadir grupo: *groupadd nombre_grupo*

En las distribuciones basadas en Debian también está disponible el comando *addgroup*.

B) Eliminar grupo: *groupdel nombre_grupo*

Si algún usuario tiene ese grupo como grupo primario, entonces no se eliminará el grupo. En las distribuciones basadas en Debian también está disponible el comando *delgroup*.

C) Modificar grupo:

groupmod [-g nuevo_GID] [-n nuevo_nombre_grupo] nombre_grupo

Permite modificar el nombe de un grupo (opción *n*) o el GID (opción *g*) del mismo.

En el sistema existen algunos grupos especiales que sirven para controlar el acceso de los usuarios a distintos dispositivos. Algunos de estos grupos son:

- *cdrom*: dispositivo CD-ROM.
- *floppy*: unidades de diskette.
- *dialout*: puerto serie.
- *audio*: controla el acceso a dispositivos relacionados con la tarjeta de sonido.

Para dar acceso a un usuario a uno de estos servicios, basta con añadirlo al grupo adecuado usando el comando *adduser* (ser verá en el siguiente apartado).

1.2. Gestión de Usuarios

Algunos de los comandos que permiten gestionar los usuarios son:

A) Añadir usuario:

useradd [-g grupo_principal] [-d directorio_inicio] [-m] [-c comentario] [-s Shell] nombre_usuario

Permite crear un usuario con la información proporcionada como parámetros. Entre estos parámetros caben destacar:

-g grupo_principal Este grupo debe existir previamente

-d directorio_inicio Se especifica la ruta completa hacia el directorio de inicio del nuevo usuario.

-m Si el directorio de inicio no existe se crea. Si no se usa esta opción no se crea el directorio de inicio para el usuario, teniendo que crearlo manualmente.

-c comentario Se asocia el comentario al usuario creado.

-s Shell Intérprete de comandos predeterminado asociado al usuario.

Si no se especifica nigún parámetro, por defecto, se crea un grupo con el nombre del usuario y éste será el grupo principal del usuario.

Por ejemplo, si se desea añadir un usuario y asignarle como grupo principal un nuevo grupo con el mismo nombre que el del usuario, el comando sería:

useradd -d /home/nuevoUsuario -m -s /bin/bash nuevoUsuario

Una vez creado el usuario queda por asignarle una contraseña con el comando *passwd* (debe ser ejecutado por el *root*). *passwd nombre_usuario*

El sistema solicitará dos veces la contraseña que se desea asignar al usuario.

B) Eliminar usuario: *userdel [-r] nombre_usuario*

Con la opción *r* se elimina también su directorio de inicio.

C) Modificar usuario:

usermod [-g grupo_principal] [-d directorio_inicio] [-l nuevo_nombre_usuario] [-c comentario] [-s Shell] nombre_usuario

Permite modificar uno o varios de los datos asociados al usuario mediante sus parámetros. Entre estos parámetros caben destacar:
-g grupo_principal Modifica el grupo principal del usuario que debe existir previamente. *-d*

directorio_inicio Modifica el directorio de inicio del usuario.

-l nuevo_nombre_usuario Cambia el login o nombre de usario.

-s Shell Cambia el intérprete de comandos predeterminado del usuario. *-c comentario* Añade o

modifica el comentario del usuario

D) Añadir usuario a un grupo: *usermod -a -G nombre_grupo nombre_usuario*

En las distribuciones basadas en Debian también está disponible el comando *adduser,* cuya sin*tax*is es*: adduser nombre_usuario nombre_grupo*

E) Quitar usuario de un grupo: *usermod -G lista_grupos nombre_usuario*

Si el usuario actualmente es miembro de algún grupo que no se especifica en la lista, automáticamente ese usuario dejará de pertenecer a ese grupo.

En las distribuciones basadas en Debian también está disponible el comando *del**user,* cuya sin*tax*is es*: deluser nombre_usuario nombre_grupo*

2. Comando sudo

2.1. Comando su

El comando **su** corresponde a las siglas de *Switch User* y sirve para cambiar de usuario sin necesisdad de hacer un cierre o cambio de sesión. Su sintaxis es:

su nombre_usuario

El sistema pedirá la contraseña del usuario al que se conmuta.

Es preciso aclarar que el prompt se actualizará indicando el usuario al que se ha conmutado, pero no se modificará el directorio actual por el directorio inicial del nuevo usuario, al igual que tampoco se modificarán el valor de otras variables de entorno.

Si se desea realizar una conmutación de usuario completa (no sólo el prompt sino todas las variables de entorno y el directorio actual por el directorio de inicio del nuevo usuario) es necesario añadir un guión (rodeado de espacios en blanco) entre el comando *su* y el nombre de usuario al que conmutar (*su - nombre_usuario*).
Para cerrar la sesión abierta con el comando *su* y volver al usuario anterior, se ha de ejecutar el comando *exit.*

2.2. COMANDO SUDO

En Ubuntu, por seguridad, la cuenta del administrador (*root*), por defecto, está desactivada al no tener contraseña asignada. Por lo que, para ejecutar tareas administrativas existe un grupo de usuarios denominado **sudoers users** (administradores), los cuales pueden obtener permisos de root, mediante el comando *sudo*. El usuario con el que se instale Ubuntu, se encuentra incluido en este grupo de administradores.

El comando **sudo** (*SUperuser DO*) permite a los usuarios ejecutar acciones con los privilegios de seguridad de *root* (o de cualquier otro usuario), es decir; si un usuario normal desea ejecutar un comando de *root* (o de cualquier otro usuario), el comando *sudo* verifica en su lista de permisos y si está permitido la ejecución de ese comando para ese usuario, entonces *sudo* se encarga de ejecutarlo. La lista de control que usa *sudo* está en */etc/sudoers*.

La sintáxis de este comando es: *sudo comando*

donde *sudo* requiere que el usuario se autentifique introduciendo su contraseña para permitirle la ejecución del comando.

Por defecto, después de introducir la contraseña, el usuario tendrá unos minutos para volver a usar el mismo comando u otros para los que tiene permiso (mediante el comando *sudo*) sin necesidad de volver a ingresar su contraseña. Si el usuario ya ha terminado de lanzar todo lo que requiere del control del comando *sudo*, puede finalizar este tiempo de gracia de validación ejecutando el comando *sudo* con la opción *k* (kill) de la siguiente forma: *sudo -k*

Si se desea conocer los comandos para los que un usuario tiene permiso, éste debe ejecutar el comando *sudo* con la opción *l* (*sudo -l*). Pero si lo que se desea es conocer los comandos para los que otro usuario distinto tiene permiso, además de usar la opción *l* se tendrá que usar la opción *U* especificando el nombre de dicho usuario (*sudo -l -U nombre_usuario*).

Es posible que se necesite ejecutar muchos comandos como *root*, entonces se puede conmutar al usuario *root*, para así, no tener que escribir el comando *sudo* en cada línea de órdenes. Existen varias formas de hacerlo:

- ✔ *sudo su*

  La contraseña que pide este comando es la del usuario que ejecuta el *sudo* no la del *root*.

  Usando el comando *su* (sin argumentos), se va a mantener el valor de la mayoría de las variables de entorno así como el valor del directorio actual. Para saber realmente el login con el que el sistema reconoce al usuario, se podría ejecutar el comando *whoami* .

- ✔ *sudo su -*

  Al usar el guión del comando *su*, se actualizan todas las variables de entorno así como el directorio de inicio.

Tanto con “sudo su” como con “sudo su -” se ejecutan muchos procesos innecesarios, y lo más importante, todos como *root*. Es por eso, que la forma más recomendable de conmutar al usuario *root* es usando la opción *i* del comando *sudo (sudo -i*). Este comando carga una shell del *root*, y lo más importante, elimina las variables de entorno del usuario cargando las del *root*.
En cualquier caso, para volver al usuario anterior (cerrar sesión de *root*) se tiene que ejecutar el comando *exit*.

2.3. Comando visudo

Este comando permite editar el fichero de configuración de *sudo sudoers,* invocando al editor que se tenga por defecto (generalmente el editor vi). Para lanzar este comando se debe tener privilegios de administrador, por lo que la sintaxis a seguir será: *sudo visudo*

En este fichero de configuración se especifica qué usuarios pueden ejecutar qué comandos en nombre de qué otros usuarios. Como *sudo* es muy estricto con el formato de este fichero, y cualquier error podría causar problemas serios, existe este comando *visudo*; que sirve para comprobar que el fichero *sudoers* no está siendo utilizado desde otra sesión del usuario *root*, evitando de esta forma la multiedición con posibles corrupciones del fichero. Además, el comando *visudo* bloquea este fichero de tal manera que nadie más lo puede utilizar, evitando que dos o más usuarios administradores modifiquen accidentalmente los cambios que el otro realizó.

Otra característica del comando *visudo* es que al cerrar el fichero, verifica que esté bien configurado, es decir, detecta si existen errores de sintaxis principalmente. En el caso de que así fuese, mostraría un mensaje de error con la línea dónde se encuentra y la pregunta “What now?”. Existen tres posibles respuestas:

- *e* : edita de nuevo el fichero colocando el cursor en la línea de error;
- *x* : salir sin guardar los cambios;
- *Q* : salir y guardar los cambios.

2.4. Fichero sudoers

El fichero *sudoers* generalmente está ubicado en */etc* y en él, como se ha comentado antes, se establece quién (usuarios) puede ejecutar qué (comandos) y de qué modo (opciones), generando efectivamente una lista de control de acceso para el comando *sudo.*

Es posible dividir el contenido de este fichero de configuración en tres partes, que por extraño que parezca, ninguna de ellas es obligatoria ni tampoco deben aparecer en algún orden específico, pero la que al menos debe de existir es la tercera, que es la definición de los controles o reglas de acceso.

A continuación, se van a describir, a grandes rasgos, cada una de estas tres partes:

(1) **Alias** El alias engloba bajo un solo nombre una serie de elementos (usuarios, comandos o equipos) que después en la parte de definición de reglas serán referidos aplicados bajo cierto criterio. La sintaxis para crear un alias es:

tipo_alias NOMBRE_ALIAS = elemento1, elemento2, ..., elementoN

El *tipo_alias* define el tipo de los elementos, es decir, dependiendo del tipo de alias serán sus elementos.

(2) **Defaults** (Opciones) Permiten definir ciertas características de comportamiento para los alias previamente creados, para usuarios, usuarios privilegiados, para equipos o de manera global para todos.

(3) **Reglas de acceso** Definen qué usuarios ejecutan qué comandos bajo qué usuario y en qué equipos.

Para comprender mejor el contenido del fichero de configuración *sudoers*, la siguiente figura muestra un posible contenido de dicho fichero junto con comentarios escritos en negrita.

 # El carácter almohadilla precede a un comentario
/etc/sudoers
#
This file MUST be edited with the 'visudo' command as root.
See the man page for details on how to write a sudoers file.

 # La siguiente opción (*env_reset*) modifica el comportamiento del comando *sudo*, en concreto va a reinicializar las variables de entorno que el comando *sudo* utiliza, quedando disponibles sólo las variables LOGNAME, SHELL, USER y USERNAME Defaults env_reset

Host alias specification

User alias specification

Cmnd alias specification

 # El usuario *root* tiene todos los privilegios, es decir puede utilizar todas las terminales, actuando como todos (cualquier) usuario y ejecutar (todos) los comandos del sistema. La sintaxis usada es:
nombre_usuario terminal = (usuarios) comandos
en este caso el nombre de usuario es *root*, el primer ALL se refiere a la terminal desde la que el usuario especificado puede ejecutar el comando *sudo*, el ALL entre paréntesis indica como que usuarios puede actuar el usuario indicado inicialmente y, por último, el último ALL especifica que comandos el usuario indicado al inicio puede ejecutar.

User privilege specification
root ALL=(ALL) ALL

 # Se permite a los miembros del grupo *sudo* ejecutar cualquier comando # Allow members of group sudo to execute any command
(Note that later entries override this, so you might need to move # it further down)
%sudo ALL=(ALL) ALL
#
#includedir /etc/sudoers.d

 # A los miembros del grupo *admin* se les asignan todos los privilegios, es decir pueden ejecutar cualquier comando, desde cualquier terminal, actuando como cualquier usuario, tal como el administrador *root*.
Members of the admin group may gain root privileges
%admin ALL=(ALL) ALL