¿Cómo administrar sus usuarios y grupos en Linux?

Actualizado el 30 de agosto, 2016. Por BlueHosting.

Por definición Linux es un sistema multiusuario; esta característica permite proporcionar servicio y procesamiento a varios usuarios de forma simultánea. Por esta razón, manejar correctamente sus usuarios y grupos es un tópico esencial en la administración de sistemas Linux. En este tutorial incluimos conceptos como: añadir y borrar usuarios/grupos, añadir usuarios a grupos específicos y le brindamos consejos importantes que le ayudarán a una administración eficiente y óptima de sus usuarios y grupos.

Los comandos mencionados durante este tutorial asumen que usted es el usuario root. Si tiene privilegios de superusuario utilice el prefijo **sudo** *antes de cada comando.*

Administración de usuarios

Crear un nuevo usuario

Para agregar un usuario nuevo, puede ejecutar alguno de los siguientes comandos:

```
adduser nuevousuario useradd nuevousuario
```

Tenga en cuenta que por defecto, esta acción creará un grupo del mismo nombre (**nuevousuario**), es decir que el identificador de usuario UID y el identificador de grupo GID serán iguales. Además se creará el directorio principal del usuario: **/home/nuevousuario** (por defecto).

Si quiere incluir al usuario en un grupo justo al momento de crearlo puede utilizar:

```
useradd nuevousuario -g grupoprincipal
```

El parámetro **-g** añade el grupo principal del nuevo usuario. ¡Recuerde que el grupo especificado debe existir! ¿Qué ocurre entonces si nuestro usuario debe estar en más de un grupo? También tiene la opción de agregar grupos secundarios usando la opción **-G** seguido

del grupo secundario. Si requiere más de un grupo secundario, puede agregar dos o más grupos separándolos por comas. Por ejemplo:

```
useradd nuevousuario -g grupoprincipal -G gruposecundario1,gruposecundario2
```

Otra opción útil de este comando es cuando se especifica un directorio principal distinto:

```
useradd -d /home/otrodirectorio
```

Para llevar un control más claro de cada usuario, y debido a que el nombre de usuario podría no proporcionar información suficiente, puede especificar el nombre completo del usuario usando la opción **-c** , por ejemplo:

```
useradd -c "Julia Schneider" julia
```

Establecer contraseñas

Una vez que cree un nuevo usuario (o en cualquier momento que considere necesario), es hora de crear una contraseña asociada a dicho usuario. Siempre se recomienda usar contraseñas con un nivel de seguridad alto:

Combine letras mayúsculas, minúsculas y números.
No use secuencias de números.
Agregue signos como **/** , ***** , **-** , **+** , **_** , **?** , **.** .

El comando utilizado para establecer contraseñas es **passwd** seguido por el nombre del usuario. Por ejemplo, para establecer la contraseña del usuario **julia** utilice:

```
passwd julia
```

Su shell le pedirá que indique la contraseña y que luego la confirme. Por seguridad Linux no muestra su contraseña ni caracteres ocultos mientras escribe, así que debe hacerlo con cuidado. A continuación un ejemplo de la salida en su terminal al usar el comando anterior:

```
Changing password for user julia. New password: Retype new password: passwd: all authentication tokens updated successfully.
```

Existen opciones que puede utilizar junto con el comando **passwd** para bloquear la contraseña (**-l** o **--lock**), desbloquearla (**u** o **--unlock**) e incluso para establecer el tiempo máximo de vida de la contraseña: **--maximum=DÍAS** .

Por ejemplo, si desea que la contraseña del usuario **julia** expire en 90 días:

```
passwd --maximum=90 julia
```

Sabrá que el procedimiento fue exitoso si ve un mensaje similar al siguiente:

```
Adjusting aging data for user julia. passwd: Success
```

Otras operaciones

Comando usermod

Con el comando **usermod** se pueden realizar varios cambios en la configuración de los usuarios del sistema. Algunos ejemplos a continuación.

Para cambiar el nombre de inicio de sesión de un usuario:

```
usermod –l nombreantiguo nombrenuevo
```

Para cambiar el directorio **/home** de un usuario utilice:

```
usermod –d /nuevo/directorio –m nombredeusuario
```

Para agregar el usuario a otros grupos suplementarios:

```
usermod –G grupo1,grupo2,grupo3
```

Use el parámetro **--expiredate** seguido por la fecha en formato AAAA-MM-DD para establecer la fecha de expiración de la cuenta de usuario:

```
usermod --expiredate 2017-01-02 nombredeusuario
```

Para bloquear la contraseña de un usuario:

```
usermod --lock nombredeusuario
```

Para desbloquear la contraseña de un usuario:

```
usermod --unlock nombredeusuario
```

Borrar una cuenta de usuario

Para borrar una cuenta de usuario, debe utilizar el comando **userdel** . Recuerde que debe ser usuario *root* para ejecutar la acción:

```
userdel nombredeusuario
```

El comando arriba borraría únicamente la cuenta del usuario **nombredeusuario** , pero no elimina los archivos asociados a este usuario.

Utilice el siguiente comando con cuidado: si desea borrar la cuenta de un usuario, y además forzar la eliminación del directorio principal y de todos sus archivos (incluso si el usuario está activo o los archivos están en uso):

```
userdel –rf nombredeusuario
```

Mostrar los grupos a los cuales pertenece un usuario

Puede usar los siguientes comandos para identificar los grupos a los cuales pertenece un usuario en particular:

```
groups nombredeusuario id nombredeusuario
```

Comando chage

El comando **chage** (change age) cambia el número de días entre los cuales debe cambiar la contraseña. Cuando el usuario inicie sesión, aparecerá un mensaje indicando que debe cambiar la contraseña antes de que expire.

Si su servidor no posee la utilidad chage puede instalarla usando:

```
apt-get install chage
```

Para Debian/Ubuntu.

```
yum install chage
```

Para RHEL/Fedora/CentOS.

Para ver información con respecto a la contraseña de un usuario ejecute:

```
chage --list nombredeusuario
```

A continuación un ejemplo de salida para un usuario cuya información de contraseña no ha sido editada:

```
Last password change : Jul 27, 2016 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
```

Para establecer el número de días máximo para la expiración de la contraseña utilice la opción **-M** seguido del número de días. Por ejemplo: digamos que quiere que la contraseña expire cada 90 días:

```
chage -M 90 nombredeusuario
```

Esto es equivalente al comando utilizado con **useradd** anteriormente.

Para establecer una contraseña como vencida o expirada, utilice el siguiente comando. Este es especialmente útil cuando usted es el administrador de un sistema Linux, y desea mantener la privacidad del usuario en cuestión.

```
chage -d 0 username
```

De esta manera, la próxima vez que el usuario inicie sesión, se le pedirá que cambie su

contraseña. Establecer fechas de expiración de contraseñas es una buena práctica que se recomienda aplicar a todos los usuarios del sistema.

Archivos importantes

/etc/passwd

Es la base de datos de usuarios en el sistema. Toda la información de usuarios locales se almacena en texto plano bajo este archivo. Cada línea representa un usuario y tiene siete campos separados por el signo de dos puntos (`:`).

```
Cuenta (nombre de usuario) : Contraseña : UID (ID de usuario) : GID (ID de grupo) : GECOS (campo opcional con propósitos informativos) : Directorio (directorio principal o home del usuario) : shell (ruta al shell predeterminado, campo opcional)
```

Para ver el contenido en este archivo puede ejecutar:

```
cat /etc/passwd
```

Se muestra una línea de ejemplo:

```
julia:x:504:506:Julia Schneider:/home/julia:/bin/bash
```

/etc/shadow

Este archivo almacena las contraseñas reales de cada usuario en formato encriptado. Además almacena cierta información relacionada con la cuenta de usuario. La explicación del contenido de este archivo se escapa del alcance de este tutorial, pero su disposición es muy similar a la del archivo **`passwd`** . Se muestra en el siguiente ejemplo una línea del archivo **`shadow`** :

```
julia:Z9B3Qve$f7tlKI8Shimc9ZDx.
7KQGFBxFUrAX2xzMEOJVJ7YcEhePU5cR8Lo09V25aFbWe51eu3047K7kTlQJ3LG1K15y1:17043:0:90:7:::
```

Administración de grupos

Linux utiliza grupos para organizar los usuarios. Estos simplemente son conjuntos de cuentas de usuarios que comparten ciertos permisos. A todos los usuarios se les asigna un identificador de usuario (uid) y de grupo (gid). Administrar correctamente sus grupos es de gran importancia.

Lista de grupos disponibles

Para enumerar los grupos disponibles en su sistema, puede utilizar el archivo **`/etc/group`** . Verá una lista de los grupos disponibles en el orden en el cual fueron agregados, los primeros de la lista son los grupos básicos del sistema:

```
root:x:0: bin:x:1:bin,daemon daemon:x:2:bin,daemon sys:x:3:bin,adm adm:x:4:adm,daemon ...
```

Los grupos del sistema están identificados con los GIDs 1-499 (identificadores de grupo reservados para el sistema). El grupo más importante es *root*, el cual otorga administración y

control total del sistema.

Crear un nuevo grupo

Para crear un grupo nuevo utilice:

```
groupadd nombredelgrupo
```

Para crear un nuevo grupo del sistema añada el parámetro **-r** al comando **groupadd** :

```
groupadd -r nombredelgrupo
```

Modificar los ajustes del grupo

El comando **groupmod** permite hacer algunos cambios a los grupos disponibles.

Para cambiar el nombre de un grupo utilice:

```
groupmod -n gruponuevo grupoantiguo
```

Agregar y remover usuarios a un grupo con gpasswd

Puede agregar usuarios a un grupo utilizando:

```
gpasswd -a nombredeusuario nombredegrupo
```

Para remover un usuario de un grupo al cual pertenece:

```
gpasswd -d nombredeusuario nombredegrupo
```

Borrar grupos

Para borrar un grupo simplemente ejecute:

```
groupdel nombredelgrupo
```

Permisos y propiedades de usuarios y grupos

A medida que crea usuarios y grupos, lo más probable es que desee determinar los permisos y propiedades de estos dentro del sistema de archivos. Este tema se escapa del alcance de este tutorial, pero se aborda en detalle en nuestra guía de Conceptos básicos de permisos y propiedades en Linux (https://docs.bluehosting.cl/tutoriales/servidores/conceptos-basicos-sobre-permisos-y-propiedades-en-linux.html)

Recursos adicionales

Este es un tema extenso que forma parte esencial de la administración de sistemas Linux;

nuestro tutorial da un punto de partida sólido para comprender los conceptos y usos básicos. Puede consultar los siguientes recursos en busca de información adicional. Aunque este material es provisto esperando que sea útil, tenga en cuenta que no podemos certificar la actualidad o precisión de los contenidos externos.

Artículo de RedHat sobre la gestión de usuarios y grupos (https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/ch-Managing_Users_and_Groups.html).

Consulte la documentación de ayuda de los comandos aquí mencionados usando la opción **`--help`** . Ejemplos: **`adduser --help`** ; **`passwd --help`** ; **`chage --help`** ,